Ad Hoc Networks 000 (2015) 1–26

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

# Security of electrostatic field persistent routing: Attacks and defense mechanisms

Oliviu C. Ghica [b], Cristina Nita-Rotaru [a], Goce Trajcevski [b,*], Peter Scheuermann [b]

[a] Department of CS, Purdue University, United States
[b] Department of EECS, Northwestern University, United States

ARTICLE INFO

*Article history:*
Received 28 October 2014
Revised 1 June 2015
Accepted 28 July 2015
Available online xxx

*Keywords:*
Security
Sensor networks
Field based routing

ABSTRACT

Electrostatic field-based routing (EFR) is a form of geographical multi-path routing where packets are routed along (a collection of) paths corresponding to electrostatic field lines defined by the charges associated with source and sink nodes. Ideally, EFR provides an efficient and scalable solution to the workload balancing problem, thereby promoting a more even depletion of the energy resources among the participating sensors. However, in addition to not being adaptable to the realistic settings that consider the actual nodes' locations, it also assumes that nodes behave in a cooperative manner. This, in turn, renders it vulnerable to various attacks.

In this article, we investigate the security aspects of EFR-based routing protocols, focusing on an instance of EFR called multi-pole field persistent routing (MP-FPR). While advancing the naïve EFR in terms of the better location-awareness energy balancing, MP-FPR is still susceptible to the same family of attacks. We provide systematic identification of the attacks that can target different components of the protocol, and propose an extended variant, secure multi-pole field persistent routing (SMP-FPR) which incorporates a collection of defense mechanisms. We present extensive experimental evaluations of the impact of the different attacks and the effectiveness of the proposed defense mechanisms.



## 1. Introduction

Wireless sensor networks (WSNs) [1] have emerged as a promising paradigm for many applications in various environments, requiring a fusion of the sensing, processing and communication tasks. In a typical WSN application, a user-initiated query is disseminated to the appropriate *source* nodes where the data of interest is locally collected. The resulting point-to-point data-stream is relayed back to a remote *sink* node which, in turn, interfaces with the user. Many routing protocols for WSN assume that nodes are location-aware and use *geography*-based (greedy) routing, according to which packets are forwarded to nodes that are physically closer to a given destination [2]. A type of geographic routing is trajectory based forwarding (TBF) [3], in which packets are routed towards the destinations along pre-defined virtual trajectories.

Electrostatic field-based routing (EFR) [4] is a multi-path routing protocol that reduces the complexity of determining and managing the collection of underlying trajectories by representing them as electrostatic field lines, rather than relying on geometric models. The field lines originate at source nodes, where the data is produced, and lead towards a designated sink node, where the data is being consumed. The main advantage of EFR is that it creates *implicitly* a collection of spatially *disjoint trajectories* for a given (source, sink) pair, which in turn enables workload balancing in dense and uniformly distributed networks by alternating the

* Corresponding author. Tel.: +847 491 7069; fax: +847 491 4455.
*E-mail address:* goce@eecs.northwestern.edu (G. Trajcevski).

http://dx.doi.org/10.1016/j.adhoc.2015.07.016
1570-8705/

packets among the available routes. However, in networks where this assumption does not hold, path-merging can occur reducing the workload balancing capabilities. Multi-pole field persistent routing (MP-FPR) protocol [5] extends EFR's applicability to less-dense and often non-uniform network distributions by actively seeking to separate any merged paths whenever network conditions allow (see [6] for more details). Another advantage of generating multiple paths via field lines based on electrostatic changes is that one can adjust the values of the charge's magnitude associated with a particular source/sink. This, in turn, enables assigning a wider/smaller geographic area, proportional to the charges' magnitude, to be allocated for the available multiple-routes between a particular (sink, source) pair.

However, MP-FPR assumes that the participating nodes always operate correctly and in a cooperative manner – an assumption no longer valid when MP-FPR is deployed in an adversarial environment. As many applications for WSNs require deployment in such an environment, it is critical to ensure that MP-FPR operates correctly even in the presence of adversaries.

In this article we focus on MP-FPR as a representative protocol for EFR routing and analyze its vulnerabilities to attacks. We study disruptions to users' data streams and the system-wide performance and resource-utilization, such as the disruption of *workload-balancing*. Our main contributions are:

- We identify a set of attacks in MP-FPR and assess their impact on the entire system. We identify a set of *control-level* attacks: *path deflection*, *path diversity deflation*, *family path intersection*, *wild-path* and *field-line hopping*, all of which are specific to electrostatic-field based routing. These attacks are carried through control messages, and can lead to quality of service degradation by disrupting the workload-balancing operation. We also identify a set of *data-level* attacks: *data denial of service* (DoS), *data pollution*, and *data stream invalidation*, which directly target users' data streams.
- We evaluate the resilience of MP-FPR to adversarial scenarios and observe the epidemic character of several attacks that can yield significant performance degradation with minimal staging efforts. For example, a *single* attack consisting of inserting eight forged charges in the system via a sink node can nearly double the standard deviation of the residual energy levels – a representative metric for describing the workload balancing performance.
- We propose and extension to MP-FPR called secure multi-pole field persistent routing (SMP-FPR), for which we analyze and compare a set of cryptographic solutions for integrity and authentication, all adapted in a manner that will retain the desiderata (in terms of operational requirements and constraints) of MP-FPR. Specifically, we compare PIKE, DS/ECC and TESLA, and conclude that TESLA [7] as the most desirable solution. Furthermore, we propose a set of novel mechanisms – k-EF, k-RPEF and PDMS – that make SMP-FPRresilient against selective forwarding of various protocol messages. The first two mechanisms, k-EF and k-RPEF, rely on multi-path in the electrostatic context, while the third one, PDMS, is a complementary monitoring scheme to provide closed-loop control over path diversity.
- We provide extensive experimental evaluations, the results of which demonstrate the effectiveness of the proposed approaches.

A preliminary version discussing selective data forwarding attacks and defenses against them appeared in EDCC 2012 [8]. This article extends [8] by providing a detailed analysis of all defense mechanisms, along with cost and feasibility analysis. We also present the details of the PDMS approach and analyze end-to-end delivery latency of the authentication/integrity mechanisms.

**Outline**: The rest of the article is organized as follows. We overview the main aspects of the MP-FPR multipath routing protocols in Section 2, followed by a detailed presentation of the adversarial model and types of attacks in Section 3. A global overview of the defense mechanisms used by SMP-FPR is presented in Section 4. Subsequently, we provide a detailed overview of the features of several cryptographic approaches that can provide integrity to SMP-FPR, along with the corresponding overhead and feasibility analysis, in Section 5. We present the resilience mechanisms against selective forwarding attacks in Section 6 and we show the results of our experimental investigation in Section 7. An overview of the related work is given in Section 8 and in Section 9 we present the concluding remarks.

## 2. Multi-pole field persistent routing

We now describe the details of the MP-FPR protocol. The assumed settings correspond to a network $\mathbf{SN} = \{sn_1, sn_2, \ldots, sn_n\}$ of $n$ wireless sensor nodes, each capable of *sensing* and *relaying*.

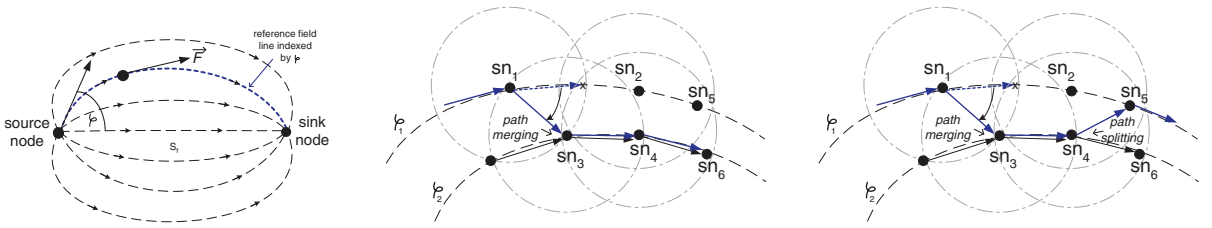
### 2.1. Overview and forwarding mechanisms

MP-FPR is based on the EFR routing protocol, which uses trajectories based on electrostatic field lines for routing. Source nodes are assigned a positive charge, and sink nodes are assigned a negative charge. To route a packet to the sink, a relay node needs to know its own location, as well as the location and the electrostatic charge information of the source and sink nodes. Permanent path deviations may occur when a given relay node cannot find subsequent relay node(s) that are along or in the vicinity of a particular electrostatic field line. Since EFR relays packets exclusively along the field line of a current relay node, paths originally following disjoint routes (i.e., different field lines) which are geographically close, may intersect/merge and overload the downstream relay nodes.

MP-FPR ensures that each packet is routed along the original field line (from which it may have been diverted) whenever possible. This, in turn, recreates spatial disjointness via splitting previously merged routes (see Fig. 1(b) and (c)), thereby ensuring better load balancing. The identity of the original field line is piggy-backed on data-packets. MP-FPR forwards messages using two mechanisms: Electrostatic Field (EF) forwarding which relies on electrostatic fields and shortest geographical path (SGP) which is a greedy based geographical routing.

**EF forwarding**: As shown in Fig. 1(a), it is based on a discrete subset of field lines between a given (source, sink) pair,

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016




**(a)** Field line selection **(b)** Permanent merging in EFR **(c)** Un-merging in MP-FPR

**Fig. 1.** MP-FPR mechanism. (a) Family of field lines between a source and a sink; (b) Path merging: $sn_1$ is unable to reach $sn_2$ and redirects to node $sn_3$ (already servicing another route); (c) Un-merging: $sn_4$ redirects the packets back to the original route (one via $sn_1$) using $sn_5$ as the next hop.

**Table 1**
MP-FPR messages.

| Type | Flow | Functionality | Protocol | Forward | Fields of interest |
|---|---|---|---|---|---|
| QUERY | Sink → sources | Query | QD-CA | SGP | $L_{\text{src}}$, $C_e$, $N_r$, |
| UPDATE | Sink → sources | Charge update | QD-CA | SGP | $L_{\text{src}}$, $C_e$ |
| RREQ | Sources → sink | Route request | RE | EF | $L_{\text{src}}$, $C_e$, $r_i$, $t_s$ |
| ACK | Sink → sources | Route ack | RE | SGP | $L_{\text{src}}$, $r_i$ |
| DATA | Sources → sink | User data | DF | EF | $r_i$, Data |

referred to as set $S_f$ – a family of paths. Each field line in $S_f$ is defined by the value of the angle $\varphi_j$, determined by the tangent to a given field line at the source and the line segment between the source and the sink. Assuming a uniform selection of the tangential-angle from $[0, 2\pi]$, a particular tangent angle $\varphi_j$ can be chosen from a family $S_f = \{k\frac{2\pi}{N_r} | k = 1, \ldots, N_r\}$, where $N_r$ is the desired size of the family of routes $S_f$. A route built along a field line with angle $\varphi_j$ is uniquely identified by an integer – its route index, denoted $r_j$.

Every node $sn_i$ can determine the value of the tangent angle $\varphi_j \in S_f$ of the field line that it actually belongs to based on: (1) the locations and charge information of all the sources plus the sink, and (2) its own location. Once $sn_i$ receives a packet, it piggy-backs the information about the field line that the packet is supposed to be forwarded along, i.e. $\varphi_j$. A particular relay node will select as a subsequent relay node, one of its 1-hop neighbors which exhibits the smallest field line deviation $|\varphi_j - \varphi_i|$, where $\varphi_i$ represents the field line a downstream relay $sn_i$ actually resides on, and it is furthest away towards the sink (cf. [5]).

**SGP forwarding**: This is a greedy geographic routing similar to BVR [9], sending packets via a geographically shortest path towards a known physical destination. In MP-FPR nodes determine their own position via a lightweight localization service external to the routing protocol (see [10] for a survey), as well as the position of their 1-hop neighbors through a periodic location information exchange.

MP-FPR protocol consists of the following components: *query dissemination and charge allocation*, *route establishment*, and *data forwarding*. We overview each component and summarize the messages used by the protocol in Table 1.

### 2.2. Query dissemination and charge allocation (QD-CA)

This component has three goals. (1) It forwards the user query towards the source node. This is achieved through a QUERY message, which is sent by the sink via SGP forwarding towards $L_{\text{src}}$ – the location within the area where data relevant to the query should be collected from. A sensor node which is closest to $L_{\text{src}}$ will assume the role of the source for the given QUERY message and initiate its processing. (2) It disseminates electrostatic charges information, which consists of a set of (location, magnitude and expiration) information associated with each routing end-point, i.e. source or sink node, in the network. For example, if there are $m$ source nodes relaying data-streams to a common sink, the QUERY message contains a set $C_e = \{e_{\text{snk}}\} \cup \{e_i | i \in 1, \ldots, m\}$ of electrostatic charges. (3) It limits the number of alternative paths to be built in order to bound the duration of the route establishment via a numerical parameter $N_r = |S_f|$ embedded in the body of the QUERY message. We refer to this limit as the *path diversity quota*.

Whenever a new data source is added to the existing set of source-nodes, a new corresponding charge is added to the virtual electrostatic field at the location of the newly identified data source. The charge information is updated at each of the source nodes via an UPDATE message. Upon receiving an UPDATE, the route establishing process is re-initiated by the source nodes in order to establish new families of routes that are consistent with the new charge distribution.

### 2.3. Route establishment (RE) and data forwarding (DF)

Initiated upon receiving a QUERY or an UPDATE at a source, *route establishment* is a two-phase request-acknowledgment process. During the *request phase*, the source transmits a set of RREQ messages along distinct electrostatic field lines towards the sink. A RREQ message carries a list of network's current charges $C_e$ as well as the field line index (equivalently route index) $r_i \in \{1, \ldots, N_r\}$ of the field line a specific RREQ is to be sent along. To amortize the transmission cost of the charges, this information is sent only once along RREQ messages, and cached locally by the relay nodes along a route; subsequent DATA messages will not carry them. The source will also incorporate its actual location information $L_{\text{src}}$ in the RREQ message such that

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

sinks maintain a more accurate representation of the actual sources. A timestamp $t_s$ is included in the RREQ message to assist in determining the quality (e.g. latency) of a specific route.

If, upon receiving a RREQ message, the route exhibited an admissible latency, the route is acknowledged via an ACK message to the specific source. The route index $r_i$ of that route is included in the ACK message. Note that ACK messages are sent back via the SGP mechanism towards the actual location of the source $L_{\text{src}}$, and not via EF mechanism the RREQ message was sent. Every acknowledged route is added to a source-maintained set of acknowledged routes $S_f^{\text{ack}} \subseteq S_f$, i.e., a pool of routes that are available for data forwarding.

**Data forwarding:** The DATA messages pertaining to a data-stream as a result of query processing are forwarded back to the sink node via the EF mechanism, by using individual routes $r_i$ from the set of acknowledged routes $S_f^{\text{ack}}$.

## 3. Taxonomy of attacks

Recall that the MP-FPR has two main objectives: (1) increase network lifetime by promoting delivery of the data stream in a workload balanced manner and (2) ensure soft QoS guarantees, such as bounded end-to-end data stream delivery latencies; both of which are offering improvements over the naïve EFR. However, each of these objectives can be affected by various attacks. In this section, we present a systematic analysis of the different components of the MP-FPR protocol and identify attacks that exploit vulnerabilities introduced by the use of electrostatic field lines and by the field persistency mechanism.

### 3.1. Adversarial model

We assume that sink nodes are not compromised and correctly follow the protocol. Further, we assume that regular nodes can get compromised, in which case they do not follow the protocol in their role as relayers. We do not consider application-level security such as ensuring the accuracy/correctness of data measurements reported by nodes acting as sources of data. This problem can be addressed by solutions complementary to our protocol, for example by having the sink ask for values from nodes in close locations and then perform voting on those values. We also do not consider denial of service attacks in which nodes in the network acting as sources of data start sending packets overwhelming the neighbor nodes or preventing other flows from succeeding in the network. While complementary techniques can be used to mitigate such behavior, ensuring multi-flow fairness and reliability in a multi-hop network with Byzantine nodes is an open problem. Honest nodes participate correctly in the routing protocol, whereas malicious nodes acting alone or in collusion can drop, delay, modify or replay packets.

We assume the forwarding mechanisms used by MP-FPR, EF and SGP, are not secure. However, since previous work examined the security of SGP [11], we focus mainly on the security of EF. Both EF and SGP rely on a localization service. We assume security mechanisms [12,13] are in place to protect the localization service. Similarly, we assume that the time synchronization mechanism is also secure [14,15].

### 3.2. Attack classification

We classify the attacks as *data-level* and *control-level* based on their target, the user-data or the network operation, respectively. An attacker can drop (or selectively forward), delay, or modify any of the five type of messages MP-FPR relies on: QUERY, UPDATE, RREQ, ACK, and DATA. We do not consider replay-attacks as they can be easily addressed by using packet sequencing or timestamps. Table 2 summarizes the attacks that MP-FPR is susceptible to.

### 3.3. Attacks during query dissemination and charge allocation

Attacks during the query dissemination and charge allocation phases can be mounted using the QUERY and UPDATE messages (cf. Table 1): *data DoS*, *data stream invalidation*, *path diversity deflation*, *path deflection*, and *family path intersection*.

**Data DoS (DoS)**: This attack can be mounted by maliciously dropping QUERY messages and disrupting the delivery of users' data-flow. The absence of the entire data-stream can be easily detected and thus the underlying attack unveiled.

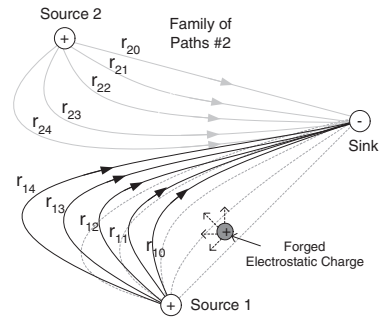
**Data stream invalidation (DSI)**: An attacker can cause a user to receive a different data than he requested by altering the $L_{\text{src}}$ parameter in the body of the QUERY message. Unlike *Data DoS*, this is a stealthy attack: user receives an uninterrupted *invalid* data stream.

**Path diversity deflation (PDD)**: This attack targets the load-balancing by reducing the number of alternate paths that the protocol can use – i.e., $N_r$ in the QUERY message. Decreasing $N_r$ reduces path diversity – e.g., maliciously setting to $N_r = 1$, will effectively degrade MP-FPRto *single*-path

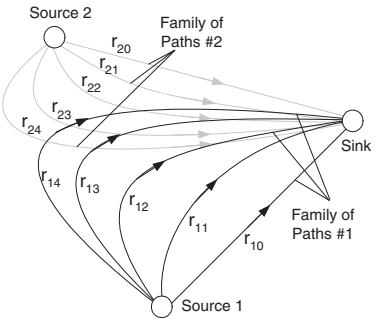
**Table 2**
Data level and control level attacks.

| Data attack | Drop | Delay | Modify |
|---|---|---|---|
| Data DoS | QUERY, DATA, ACK | DATA | DATA($r_i$), ACK($r_i$) |
| Data pollution | – | – | DATA(payload) |
| Data stream invalidation | – | – | QUERY($L_{\text{src}}$) |
| **Control ttack** | Drop | Delay | Modify |
| Path deflection | – | – | QUERY($C_e$), UPDATE($C_e$) |
| Path diversity deflation | RREQ, ACK | RREQ | QUERY ($N_r$), ACK($r_i$, $L_{\text{src}}$), RREQ($r_i$, $L_{\text{src}}$, $t_s$) |
| Family path intersection | UPDATE | UPDATE | QUERY($C_e$), UPDATE($L_{\text{src}}$, $C_e$) |
| Wild path | – | – | RREQ($C_e$) |
| Field-line hopping | – | – | RREQ($r_i$), DATA($r_i$) |

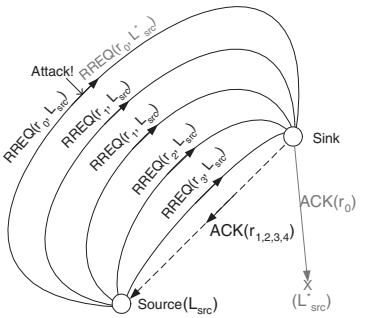
Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016



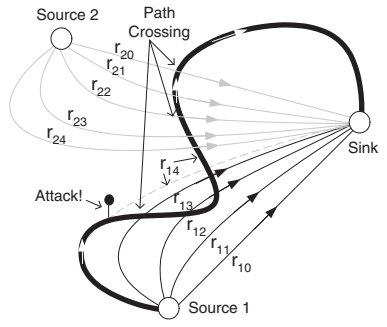

**(a) Path deflection**. Routes $r_{10}$ through $r_{14}$ will be detoured around the area where the added forged charge resides.



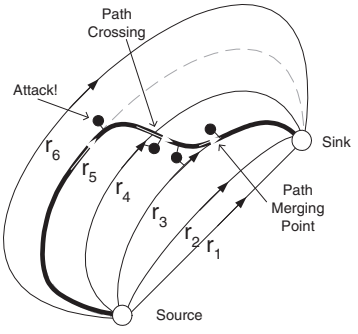

**(b) Family path intersection**. If source 1 is unaware of the charge associated with source 2, $S_{f_1}$ routes will intersect $S_{f_2}$.



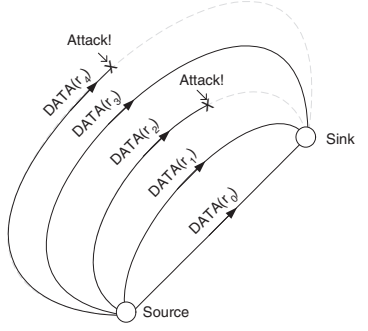

**(c) Path diversity deflation**. Path $r_0$ will never be used for data-forwarding since it is never acknowledged to source $L_{src}$.




**(d) Wild path**. Path $r_{14}$ connecting source 1 to the sink, deviates from its original route and intersects with other paths.




**(e) Field line hopping**. If $r_5$ is altered to $r_3$, then original $r_5$ deviates, intersecting $r_4$ before it merges with $r_3$, doubling the load from the merging point on $r_3$.




**(f) Data DoS**. Selective forwarding of DATA messages sent along routes $r_2$ and $r_4$ cause the user to receive an incomplete data-stream at the sink node.

**Fig. 2.** Examples of attacks against the MP-FPR protocol.

routing. *Path diversity deflation* may not have an immediate, noticeable impact to the user, however, its damaging effect is visible through a significant reduction of network's lifetime.

**Path deflection (PD)**: This attack causes a geographical shift of the existing families of routes, or a constraint in the field-region for path building. It can be conducted by modifying/forging the charge information in either the QUERY or UPDATE messages. Altering the magnitude of a charge will affect the load-balancing among distinct families of routes. In extreme cases, it is possible to narrow the admissible relay field so much that most of the paths will merge, leading to a single-path routing behavior equivalent to the *path diversity deflation* attack. Adding one forged charge may result in a geographical shift of the existing families of routes, possibly leading to increased routes' lengths, thereby increasing the end-to-end delivery latencies. Fig. 2(a) presents a family-path geographical shift as a result of one forged charge.

**Family path intersection (FPI)**: This attack targets the disjointness of the routing paths from the same family, or from distinct families. The attack can be mounted by either dropping UPDATE messages or by modifying the $L_{\text{src}}$ parameter in the UPDATE message. Some of the conditions that lead to a path deflection may also create intersections between routes pertaining to different families if charge information becomes inconsistent among families. Paths pertaining to the same family will continue to maintain the non-intersection property among themselves, however, distinct families of routes will cross their geographical bounds. Such intersections create resource utilization hot-spots with direct consequences on the overall network's lifetime. Fig. 2(b) illustrates a family path intersection attack.

### 3.4. Attacks during route establishment

The attacks in this phase target RREQ and ACK control messages respectively. We analyze attacker strategies and, in addition to the *path diversity deflation* and *data DoS*, we identify two new attacks: *wild path* and *field line hopping*.

**Path diversity deflation (PDD)**: Dropping either RREQ or ACK messages may result in an overall reduction of the route content within a family of routes. Since paths are designed to spread through a larger network-area for workload balancing purposes, an attacker can target an arbitrary node, without a priori insider information. Additionally, delaying RREQ messages or altering the embedded source-transmission timestamp $t_s$ may increase the latency beyond a user-defined tolerance. Changing the source location information $L_{\text{src}}$ in the RREQ or ACK messages will cause ACK messages to be delivered to a node different than the source. Finally, altering route index information $r_i$ in the RREQ or ACK messages can also lead to the same outcome. For example, in either case, the (corresponding) acknowledgment will

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

acknowledge an arbitrary route, which may have been already acknowledged, while the intended route will be dropped from usage. Fig. 2(c) presents an example of a *path diversity deflation* attack where route $r_0$'s acknowledgment is never received by the originating source node. All of these conditions can ultimately lead to diminished energy consumption balancing performance.

**Data DoS (DDoS)**: This attack can be mounted by targeting ACK messages. During the *route acknowledgment* phase, compromising ACK messages vs. RREQ messages can lead to different effects, because distinct forwarding mechanisms handle the two types of messages: ACKs are sent via a single-path (SGP), whereas RREQ via EF. Thus, if a single node along the SGP path is compromised, *all* ACK messages can be compromised or dropped. Since path diversity can be effectively reduced to zero, the user's data-stream will be completely blocked. Alternatively, a malicious node may alter the route index $r_i$ in the ACK message. In this case, an arbitrary route will receive an acknowledgment, possibly one that was not probed or one that may not satisfy user requirements, such as end-to-end delivery latency.

**Wild path (WP)**: The effect of this attack is to cause a route from a given family of routes to break the disjointness property of electrostatic field lines and start intersecting other routes. There are two differences from the *family path intersection* attack: (1) a wild path attack targets a single route, rather than an entire family of routes, and (2) the compromised route intersects not only other routes within the same family, but also routes pertaining to other families. This attack is carried by altering the *charge* information within a relay node along a particular route. Recall that charge information transmitted via RREQ messages are cached by the relay nodes for subsequent use. Consequently, the attack can be carried by altering the RREQ messages before their contents are cached. The entire path downstream of the compromised node will exhibit an abrupt deviation from the designated field line. Fig. 2(d) shows an example of a *wild path* attack.

**Field line hopping (FLH)**: Consider a route indexed by $r_j$, which is built along a reference field line $\varphi_j$. If the route index from in the RREQ message is altered, the original route will suddenly change its reference field line and "hop" to a different one within the same family. The immediate consequence is path intersection or merging. This situation is different from a *wild path* situation, because field lines do not change; rather, the actual route changes field lines. Fig. 2(e) shows an example of *field line hopping* attack. Field line hopping creates relay node overload, resulting in degraded balance of the energy consumption among the nodes and reduction of lifetime expectancy.

### 3.5. Attacks during data forwarding

DATA messages carry the information-load resulting from processing a user-submitted query. Since DATA messages follow probed and acknowledged paths, they are susceptible to the same attacks as those against RREQ messages.

**Data DoS (DoS)**: This attack blocks a user data-stream. It can be mounted by selectively dropping DATA messages along a path, i.e. if one of the relay nodes along the path is compromised. Fig. 2(f) illustrates this scenario, showing the effects of two different compromised nodes along different routes dropping all incoming DATA messages. In some instances, altering the route index information $r_i$ in the DATA messages, which can redirect the message along non-probed and possibly long paths, or simply delaying these messages, may similarly lead to a *data DoS* attack. In both cases, it is likely that the message will be discarded at the sink node if not received within certain admissible delay tolerances.

**Field line hopping (FLH)**: Analogous to attacks carried through RREQ messages, DATA messages can be maliciously "re-routed" along different routes than the originally prescribed ones, resulting in path merging and overloading of some of the downstream relay nodes. The net effect consists of energy consumption balancing disruption and a reduction of network's lifetime. This attack can be achieved by modifying the route index $r_i$ embedded in the DATA message.

**Data pollution (DP)**: The attacker may directly alter the user-payload within the DATA message itself. This attack can be severe, since the user may not be able to distinguish valid data from faux, and it may require advanced data analysis to detect anomalies in the data-stream.

## 4. Defense overview

We now discuss in detail the assumptions used when designing defense mechanisms. We first overview the global settings and the features of different types of motes. Subsequently, categorize the causes for the attacks identified in Section 3. Finally, we categorize the defense techniques, identifying three authentication and integrity mechanisms, and three resilience mechanisms to secure SMP-FPR against the attacks.

### 4.1. Assumptions

We design defense mechanisms defining the SMP-FPR protocol in with the objective that SMP-FPR will not reduce the scope and applicability of the originally envisioned MP-FPR multipath protocol. In other words, the defense-enabling features of SMP-FPR need to fully comply with MP-FPR's system settings: (1) very large sensor networks typically consisting of thousands of nodes, and (2) possibly non-uniform network distributions of various densities.

Furthermore, any solution aiming to become a part of SMP-FPR needs to account for the resource limitations of real motes, such as memory and processing capabilities. We evaluate the candidate solutions against several popular mote platforms: Imote2, Mica2Dot, MicaZ, TelosB [16] and Tmote Sky [17]. A summary specification of the relevant parameters of these sensors is outlined in Table 3. We note that, with the exception of the small-sized *Mica2Dot*, which is representative for large-scale distributions, the selection of the motes is consistent with the one made in [18], where an actual implementation of a cryptographic solution on various platforms is tested.

The SGP and EF message forwarding mechanisms in SMP-FPR require a separate, lightweight and trusted localization service. In this work, we assume that a localization service meeting these criteria is readily available, as existing works have thoroughly addressed this problem [10,19,20]. A secure time synchronization service is required to maintain time consistency across the entire network – since SMP-FPR

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

**Table 3**
Platform specifications.

| Platform | Voltage [V] | Current drawn CPU active [mA] | Current drawn TX [mA] | Current drawn RX [mA] | Data rate [kbps] | Program memory FLASH [KB] | RAM SRAM [KB] | ROM EEPROM [KB] |
|---|---|---|---|---|---|---|---|---|
| Mica2Dot | 3.0 | 8 | 27.0 | 10.0 | 38.4 | 128 | 512 | 4 |
| MicaZ | 3.0 | 8 | 17.4 | 19.7 | 250.0 | 128 | 512 | 4 |
| TelosB | 3.0 | 1.8 | 27.0 | 23.0 | 250.0 | 48 | 10 | 1024 |
| Tmote Sky | 3.0 | 1.8 | 19.5 | 21.8 | 250.0 | 48 | 10 | 1024 |
| Imote2 | 4.5 | 31 | 44.0 | 44.0 | 250.0 | 256 | 32,000 | 32,000 |

**Table 4**
Categorization of causes for attacks.

| Type of message | Type of attack | Defense approach |
|---|---|---|
| Control | Modify message | Authentication and integrity cryptographic mechanisms |
| Data Forward | Drop or delay message | Redundancy in the forwarding mechanism |

relies on temporal dimension in order to estimate the quality of paths by time-stamping certain protocol messages (e.g., RREQ messages). We rely on solutions such as [14,15] to provide security guarantees over the time synchronization services. We assume that the localization and time synchronization services are robust to abuses towards resource depletion via link and physical layer jamming [21].

### 4.2. Overview of the defense mechanisms in SMP-FPR

In Section 3 we gave a detailed overview of the different kinds of attacks to which components of MP-FPR may be susceptible and SMP-FPRis aiming to cope with. However, from a global perspective, there are two fundamental categories of enabling causes for those attacks, illustrated in Table 4. As shown, when the message is of a type *control* or *data*, the attacks are targeting the modification of its content. To cope with this, SMP-FPR needs *authentication* and *integrity* mechanisms. If, on the other hand, the forwarding component of the message is affected by an attack – which can be either via selective forwarding (dropping) or delaying it – SMP-FPR can handle it by providing redundancy in the forwarding mechanism, which reduces the likelihood of dropping all copies of a given message.

Based on the above classification, we identify two categories of approaches that are fundamental for SMP-FPR:

(1) *Authentication and integrity:* these can be provided with the existing cryptographic approaches such as symmetric key-based *HMAC* [22], public-key based *digital signatures* [23,24], or a hybrid solution like *TESLA* [7]. We provide a detailed analysis of three message authentication and integrity mechanisms – PIKE, DS/ECC, and TESLA – in Section 5, followed by assessing the trade-offs between security properties and costs in Section 5.2. They are primarily considered to address the attacks carried out via message-forging as outlined in Table 2, by enabling nodes to detect and filter out modified messages. Additionally, they enable detection of adversarial activity for which isolation mechanisms can be employed. Specifically, path deflection, diversity deflation, family path intersection and wild-path carried through forging electrostatic charges in QUERY, UPDATE or RREQ messages can be prevented. Moreover, field-line hopping, data DoS, data pollution, and data stream invalidation can be prevented as well by authentication and integrity mechanisms.

(2) *Resilience mechanisms to improve robustness:* In Section 6 we present three mechanisms used in SMP-FPR to improve robustness of MP-FPR to attacks carried through selective forwarding of the respective protocol messages: k-EF, k-RPEF (reverse-path k-EF), and PDMS. The k-EF represents a multi-path resilient variant of the EF forwarding mechanism, designed to provide defense against data DoS attacks. The k-RPEF aims at replacing the SGP mechanisms with the EF for handling QUERY, UPDATE and ACK messages, in order to provide adequate protection against path diversity deflation, family path intersection and certain data DoS attacks. The path diversity monitoring scheme (PDMS) is a reactive defense mechanism against path diversity deflation attacks. PDMS is designed to complement the k-RPEF defense solution by providing a close-loop control mechanism for ensuring adequate path diversity in an adversarial context. We note that the k-EF and k-RPEF robustness mechanisms can also provide the same benefits to attacks carried through delaying of MP-FPR messages, as they increase the likelihood that at least one instance of a given message reaches the destination on time.

A high level summary-comparison of some of the features of the defense mechanisms employed in SMP-FPR is presented in Table 5. Due to its low overheads in terms of

**Table 5**
Feasibility of defense mechanisms.

| | PIKE | DS/ECC | TESLA | k-EF | k-RPEF | PDMS |
|---|---|---|---|---|---|---|
| No platform memory limitations | – | √ | √ | √ | √ | √ |
| Low communication overhead | – | √ | √ | √ | √ | √ |
| Negligible processing overhead | √ | – | √ | √ | √ | √ |
| Low overall latency overhead | – | – | √ | √ | √ | √ |
| Low energy overhead | – | – | √ | √ | √ | √ |

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

energy/communication, latency and processing, we have selected *TESLA*144 [7] as a mechanism of choice for authentication and integrity – part of the reason being that it does not induce any special architectural/hardware requirements. A detailed analysis of the features of each of the mechanisms in Table 5 across several contexts, along with the respective overheads, is presented next.

## 5. Authentication and integrity mechanisms in SMP-FPR

We now focus on the part of SMP-FPR aiming to provide solutions to authentication and integrity verification.

First, we go over the distinct features of the possible cryptographic approaches – instances of *symmetric* key cryptography (*HMAC* [22]), *public* key cryptography (*digital signatures* [23,24]), and a hybrid cryptographic solution (*TESLA* [7]). We discuss how each of them could achieve *secure path establishment* where authentication and integrity are provided between end-points and *hop-by-hop authentication* where such services are provided on a hop-by-hop basis. We also analyze the respective energy consumption overheads on a per-node basis in the following categories: (1) bootstrapping and key pre-distribution, (2) secure multi-path establishment and (3) data forwarding. Assuming the motes' specifications outlined in Table 3, we use $I_P$, $I_{Tx}$ and $I_{Rx}$ to denote the current drawn (in milli-amperes), due to internal processing, transmission, and receiving of data packets. Similarly, we use $T_P$, $T_{Tx}$ and $T_{Rx}$ to represent the duration (in milli-seconds) of performing a specific task. Given battery voltage $U$, the expression that we used for energy consumption analysis becomes $E = U \cdot (T_P I_P + T_{Tx} I_{Tx} + T_{Rx} I_{Rx})$.

Subsequently, we present a detailed *comparative* analysis of the various overheads incurred by each of the potential authentication and integrity mechanisms in terms of *memory, communication, processing* and *latency*. In these contexts, we focus on two typical phases of a sensor network deployment: (1) *bootstrapping* which captures the immediate post-deployment setup, including node discovery and initial secure topology establishment; and (2) *operational*, which defines the remaining period of effective usage.

### 5.1. Approaches to integrity and authentication

Typically, the mechanisms providing integrity and authentication are based on: symmetric key cryptography, public-key cryptography, and hybrid solutions. Below we discuss each of them in the context of the constraints imposed by sensor networks and the capabilities of sensor nodes.

*Symmetric key cryptography schemes* in the form of key pre-distribution have been the preferred authentication and integrity method for WSNs due to the prohibitive cost of public-key cryptography. Proposed approaches are: (1) single-mission key pre-distribution, (2) fully pair-wise key pre-distribution, (3) random/probabilistic key pre-distribution schemes, (4) centralized key distribution center schemes (KDC), and (5) decentralized key distribution center schemes (dKDC). Once such keys exist keyed-MACs such as HMAC can be used for authentication and integrity.

Single-mission and fully pair-wise pre-distribution are inadequate solutions. The single-mission keys scheme incurs the least overhead but provides poor resilience to attacks. Full pair-wise key pre-distribution promises the best achievable security, but introduces a scalability concern, as the memory overhead becomes $O(n)$, where $n$ is the number of nodes in the network.

Probabilistic key pre-distribution schemes address the full pair-wise scalability concerns while achieving comparable security benefits. The scheme [25] relies on probabilistic key sharing among nodes to establish an initial (connected) topology upon which localized-key sharing would be achieved, at run-time, when needed. The memory overhead is effectively reduced to $O(k)$, $k \ll n$, where $k$ represents the size of a set of keys pre-loaded on each node. The scheme has been improved, most notably in [26], providing increased resilience to attacks. Fundamentally, these approaches rely on a random-graph model, which is connected with very high probability if and only if the average degree of nodes is large [27]. For a typical range of acceptable low-connectivity risk probabilities the absolute lower bound on the node degree requirement varies between 13 and 20 neighbors per node for smaller networks of 1000 nodes, and increases to 15−22 for larger networks of 10, 000 nodes. Consequently, such a scheme will severely limit the applicability of SMP-FPRto high-density applications only, offsetting its core benefits in practical lower density networks (i.e. as low as 8 neighbors per node, on average).

KDC-based schemes (e.g. SPINS [28] and Kerberos [29]) rely on the presence of a centralized key distribution center (KDC) to act as a trusted arbiter for key establishment. As with all centralized approaches, the distribution center becomes a single point of failure for the security of the entire network. Decentralized key distribution schemes (dKDC) like PIKE (Peer intermediaries for key establishment) [27] have a reduced overhead. PIKE relies on a trusted subset of nodes to perform key-management.

*Public key cryptography solutions* became affordable in WSNs due to recent advances in sensor networks which led to an increase of computational and memory resources. The comprehensive experimental analysis in [18] gave compelling arguments for elliptic curve cryptography (ECC) public key cryptography. ECC features small key sizes and compact signatures, i.e. to provide equivalent security to 1024-bit RSA, an ECC scheme only needs 160-bit key size.

*Hybrid public/symmetric key cryptography solutions* to authentication and integrity aim at combining the benefits of symmetric and public-key schemes: the smaller computational overhead of using symmetric keys and the smaller communication overhead corresponding to public key cryptography. A well-known hybrid scheme is *TESLA* (time efficient stream loss-tolerant authentication) [7] which signs streams of data using keyed message authentication codes (MACs). TESLA uses public key cryptography to securely disseminate the initial signature.

#### 5.1.1. HMAC via PIKE

HMAC is a hash-based message authentication code which relies on secret symmetric keys. PIKE implements the key pre-distribution and establishment that enables the use of HMAC. PIKE is compatible with both low and high density networks as well as non-uniform distributions, which complies with the context under which SMP-FPR operates. PIKE devises and pre-distributes a set of $\sqrt{n}$ keys to guarantee

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

**Table 6**
Analytical analysis of energy consumption overhead for PIKE/HMAC (TinyHASH) with data-rates $R$, for hop-by-hop authentication.

| Phase | Units | PIKE/HMAC (TinyHASH) | Dominant node |
|---|---|---|---|
| Bootstrapping | [mJ/node] | $U((I_{Tx}+I_{Rx})\cdot(\kappa+\varrho)n/R)$ | N/A |
| Path-establishment | [mJ/node/path] | $U\left((I_{Tx}+I_{Rx})\cdot\frac{4}{3}(K+\varrho)N_r\alpha\sqrt{n})/R+I_PP_t\right)$ | Source node |
| Data-forwarding | [mJ/node/packet] | $U((I_{Tx}+I_{Rx})\cdot hR_L/R+I_PP_t)$ | Relay node |

**Table 7**
Analytical analysis of energy consumption overhead for ECC/ECCDSA (TinyECC) with data-rates $R$ for hop-by-hop authentication.

| Phase | Units | ECC/ECCDSA | Dominant node/Role |
|---|---|---|---|
| Bootstrapping | [mJ/node] | – | N/A |
| Path-establishment | mJ/node/path] | $U((I_{Tx}+I_{Rx})\cdot(K+\varrho)N_rR_L/R+I_PP_gN_rR_L)$ | Source node |
| Data-forwarding | [mJ/node/packet] | $U((I_{Tx}+I_{Rx})\cdot s/R+I_PP_v))$ | Relay node |

connectivity initially to a subset of nodes which form a basis for further key establishment via *intermediaries*. An intermediary shares keys with two other nodes in the network, through which a secure communication path can be established.

**Secure path establishment**: In order to establish a secure path to another node, the initiating node generates a new path-key and sends it encrypted to one (of possible two) intermediary node with whom both end-nodes share independent keys. The intermediary decrypts and re-encrypts the path-key using the other end-node's shared key, before sending it through. A nonce message is sent back to the initiating node to confirm the establishment of the path.

**Hop-by-hop authentication**: In order to provide hop-by-hop authentication in SMP-FPR we need to establish symmetric path-keys between a source and each of the relays along a route to destination. For sink-to-source secure path establishment, an additional 'SCOUT' message will be sent before MP-FPR's QUERY message. The purpose of the SCOUT message is to trigger individual symmetric key establishment between on-route relay nodes and the initiating sink node. A SCOUT-BACK message will be returned to the sink confirming the completion of the path establishment. The process is similar for the source-to-sink multi-path: it is triggered via S-RREQ messages, which will precede MP-FPR's standard RREQ messages, thus providing authentication of sensitive charge information within RREQ. To allow undistorted path-length estimation, RREQ' packet size can be increased artificially to supersede the size of the DATA packets. Corresponding ACK messages will follow the secure path established via SCOUT.

**Bootstrapping**: To enable quick discovery of intermediaries without the need of controlled flooding, PIKE employs an address lookup service such as GHT [30] where the (*id*, *location*) information of the peer intermediaries are stored. The nodes that support the GHT structure are called *replication* points. GHT establishment takes place only once, during bootstrapping phase, when information about the geo-location of the intermediaries is disseminated to the replication nodes. According to PIKE, each node will send its identity and localization information to its nearest replication node, from where it is forwarded to the "correct" replication node, which in turn is determined by hashing the identity information of the intermediary.

Table 6 presents the analytical evaluation of the energy overheads of PKIE/HMAC in terms of both the processing and the communication timings results.

#### 5.1.2. Digital signatures/ECC

Public keys can be generated by each individual node *post*-deployment, during the operational phase, in order to enable digital signature based authentication of protocol messages exchanged in the network.

**Secure path establishment**: When two end-nodes intend to establish a secure path, the originating node needs to acquire the public key of the terminus node in order to digitally sign all subsequent outgoing messages. Conceptually, this is a two step process: (1) the originating node announces its intention to establish a secure channel to the terminus node; (2) the terminus replies to the originating node with the public key to be used to perform the encryption.

**Hop-by-hop authentication**: can be easily supported by public key cryptography, requiring the same modifications to the MP-FPR protocol as for HMAC/PIKE. However, instead of triggering path-key establishment between end points and intermediary relay nodes, the SCOUT and S-RREQ messages will contain the public key of the node where the route originates. The public key is stored at the destination and cached by every relay node in between.

**Bootstrapping**: There is no intrinsic bootstrapping overhead when using ECC-based public key cryptography scheme, with the exception of the initialization and generation times of individual public-keys for each node. DS/TinyECC does not rely on any other services to operate.

A summary analysis of the per node energy-overheads of ECC/ECCDSA is presented in Table 7.

#### 5.1.3. TESLA

In TESLA the sender commits to a random key $k$ and transmits it to the receivers. The sender then uses $k$ to generate and attaches a keyed MAC to the next packet $P_i$. In a later packet $P_{i+1}$, the sender de-commits to $k$, which allows the receivers to verify the commitment and the keyed MAC of packet $P_i$. If both verifications are correct, then a receiver knows that the packet $P_i$ is authentic. To bootstrap this scheme, the sender uses a regular public signature scheme

**Table 8**
Analytical analysis of energy consumption overhead for TESLA with data-rates $R$ for hop-by-hop authentication.

| Phase | Units | TESLA (TinyECC + TinyHASH) | Dominant Node/Role |
|---|---|---|---|
| Bootstrapping | [mJ/node] | – | Replication Point |
| Path-establishment | [mJ/node/path] | $U((I_{Tx}+I_{Rx})\cdot(K+\varrho)N_r/R + I_P P_g N_r)$ | Source Node |
| Data-forwarding | [mJ/node/packet] | $U((I_{Tx}+I_{Rx})\cdot 3K/R + I_P P_v))$ | Relay node |

to sign the initial commitment, whereas all the other packets are authenticated through chaining.

**Secure path establishment**: Without loss of generality, we explain this mechanism from the DATA forwarding perspective. The path establishment process is identical with the one described in PIKE considering hop-by-hop authentication, with one difference: the path's originating node, i.e. the source node, will include in S-RREQ an initial key commitment. This allows to authenticate the entire stream of packets that will be carried and the subsequent keys and commitments within.

**Bootstrapping**: TESLA relies on TinyECC public-key mechanism for sending the initial commitments, thus the bootstrapping overhead, just as in DS/TinyECC variant, is given by the one-time generation of the public keys during TinyECC initialization step, along with the corresponding memory requirement for TinyECC implementation. TESLA does require that sensors are loosely time-synchronized.

Table 8 illustrates the energy overheads on per node bases for TESLA.

### 5.2. Analytical comparison of candidate

We now follow with a detailed comparative analysis of the overheads incurred by each of the PIKE/HMAC, digital signature/ECC and TESLA authentication and integrity mechanisms associated with *memory, communication, processing, latency* and *energy*.

#### 5.2.1. Memory overhead

**PIKE/HMAC** uses $\lceil\sqrt{n}\rceil + 1$ pre-distributed keys. Each *relaying* node needs to store one additional secret key known by itself and the initiator of the route. Given that SMP-FPRaims at retaining the disjoint paths as enabled by MP-FPR, under ideal conditions, a relay node is expected to carry messages from only *one* initiator. Thus, the total expected storage overhead is $\lceil\sqrt{n}\rceil + 2$ keys.

Considering hop-by-hop authentication, the source needs to store the shared key with the sink, and $N_r R_L$ keys, to secure each of the $(N_r)$ paths, where $R_L$ is the average hop-count of a path. Keeping the assumption of no restrictions over the location of the source and sink nodes in SMP-FPR, the expected shortest-hop distance between any two nodes is guaranteed by PIKE to be $\alpha\sqrt{n}$, where $\alpha$ is a constant dependent on the range of nodes and shape of the deployment area. Considering the hop-count ratio $\beta$ between the longest admissible alternate path and the shortest path, which models the maximal path-length query-specified restrictions in MP-FPR, path-length is expressed as $R_L = \alpha\frac{\beta+1}{2}\sqrt{n}$. However, some of the nodes already share keys with the source node with a probability of $\frac{\sqrt{n}}{n}\cdot\frac{N_r R_L}{n} = \frac{N_r R_L}{\sqrt{n^3}}$ (cf. [27]), where $\frac{\sqrt{n}}{n}$ represents the probability that two arbitrary nodes share a key and $\frac{N_r R_L}{n}$ is the probability that the respective node serves one of the $N_r$ multipaths. Thus, the effective additional memory overhead is $N_r R_L(1-\frac{N_r R_L}{\sqrt{n^3}})$.

PIKE has additional storage overhead due to the bootstrapping procedure that needs to store the localization information of intermediary nodes at GHT's replication points. In order to maintain the targeted scalability of $O(\sqrt{n})$ from the perspective of GHT's overhead, and without loss of generality, we have considered the total number of replication points in the network to be $m = \lceil\sqrt{n}\rceil$, where $n$ is the total number of sensors in the network. For this, each GHT's replication node will store an equal share of the network-wide id-location mapping. For example, assuming that $\kappa$ bits are required for identification and location information, the memory overhead of a replication node is $\kappa\frac{n}{m} = \kappa\sqrt{n}$ bits.

Assuming $K$ is the bit-size of a symmetric key, the upper bound of *per-node* memory overhead in PIKE/HMAC scheme is dictated by the source nodes and it has the following expressions: $M^{key}_{PIKE} \simeq K(\sqrt{n}+1) + KN_r R_L(1-\frac{N_r R_L}{\sqrt{n^3}}) + \kappa\sqrt{n}$.

**Digital signatures/ECC** – The memory overhead per-node is constant and independent of the number of links that need to be secured. The source's $K$-bit size public key needs to be cached at each relay node. Sink node incurs the largest overhead: given $Q_{\max}$ – the maximum number of concomitant queries the network can support, correspondingly the number of source nodes that can exist at any time in the network, the sink needs to store all $Q_{\max}$ public keys of all the source nodes. Therefore, from the sink's perspective, the total per-node memory overhead under the ECC scheme is given as: $M^{key}_{ECC} \simeq KQ_{\max}$.

**TESLA** – There is no bootstrapping overhead. Without loss of generality, we focus on DATA forwarding. The security of the path is maintained during data forwarding by piggybacking signed commitments on DATA message, using symmetric keys, to enable authentication for future messages. TESLA requires a buffer of $dR$ entries to be allocated, where $R$ represents the packet-transmission rate and $d$ represents the disclosure lag $d$. Each buffer entry consists of (1) signed commitment for a future message, (2) the symmetric key used for authentication of the previous message, (3) keyed MAC codes of the current message and (4) the current messages itself. Assuming that the signed commitment, the symmetric key and the keyed MAC codes are equally sized to $K$ bits and the payload size of the data messages is $p$, then the memory overhead can be specified as: $M^{key}_{TESLA} \simeq dR(3K+p)$.

**Practical comparative analysis**: Table 9 presents the RAM/ROM memory overhead, based on real implementations of ECC in TinyECC and respectively HMAC in TinyHash, for various network sizes and densities. We have considered the cases in which all TinyECC optimizations are either

**Table 9**
Per-node memory overhead summary; $\alpha = .5$, $\beta = 2$, $K = 160$ bits, $\kappa = 48$ bits, $Q_{max} = 10$, $p = 36B$, $r = 1$ pps, $d = 5s$. For PIKE-GHT no. of replication points is $m = \lceil \sqrt{n} \rceil$. Number of routes $N_r = 30$.

| Protocol | Net. Size $n$ | Expected Route length $R_L$ | Bootstrapping: Key predistribution (ROM) | Bootstrapping: Initialization overhead (RAM) | Operational (RAM) | Program (RAM) | Program (ROM) |
|---|---|---|---|---|---|---|---|
| | [nodes] | [hops] | [KB] | [KB] | [KB] | [KB] | [KB] |
| PIKE | 1,000 | 24 | 0.64 | 0.19 | 13.74 | 0.02 | 0.49 |
| | 10,000 | 75 | 1.97 | 0.59 | 43.85 | 0.02 | 0.49 |
| TinyECC | 1,000 | 24 | 0 | 0 | 0.20 | 0.03 | 1.22 |
| (w/o opt) | 10,000 | 75 | 0 | 0 | 0.20 | 0.03 | 1.22 |
| TinyECC | 1,000 | 24 | 0 | 0 | 0.20 | 0.18 | 1.83 |
| (w/ opt) | 10,000 | 75 | 0 | 0 | 0.20 | 0.18 | 1.83 |
| TESLA | 1,000 | 24 | 0 | 0 | 0.47 | 0.20 | 2.32 |
| | 10,000 | 75 | 0 | 0 | 0.47 | 0.20 | 2.32 |

**Table 10**
Feasibility analysis; $\alpha = .5$, $\beta = 2$, $K = 160$ bits, $\kappa = 48$ bits, $Q_{max} = 10$, $p = 36$ Bytes, $r = 1$ pps, $d = 5s$. For PIKE-GHT no. of replication points is $m = \lceil \sqrt{n} \rceil$. Number of routes $N_r = 30$.

| Protocol | Net. Size $n$ | Expected route length $R_L$ | TOTAL memory overhead (RAM) | TOTAL memory overhead (ROM) | Feasibility ('-':yes, 'x':no) Mica2Dot | MicaZ | TelosB | Tmote sky | Imote2 |
|---|---|---|---|---|---|---|---|---|---|
| | [nodes] | [hops] | [KB] | [KB] | | | | | |
| PIKE | 1,000 | 24 | 13.97 | 1.13 | – | – | x | x | – |
| | 10,000 | 75 | 44.47 | 2.46 | – | – | x | x | – |
| TinyECC | 1,000 | 24 | 0.42 | 1.22 | – | – | – | - | – |
| (w/o opt) | 10,000 | 75 | 0.42 | 1.22 | | – | – | – | – |
| TinyECC | 1,000 | 24 | 0.57 | 1.83 | – | – | – | – | – |
| (w/ opt) | 10,000 | 75 | 0.57 | 1.83 | – | – | – | – | – |
| TESLA | 1,000 | 24 | 1.14 | 2.32 | – | – | – | – | – |
| | 10,000 | 75 | 1.14 | 2.32 | – | – | – | – | – |

enabled or disabled. Table 10 cumulates the memory overhead and highlights the platforms which cannot accommodate the specific memory demand.

PIKE's memory demand is significantly higher, outweighing both ECC and TESLA by up to two orders of magnitude. Moreover, the memory demand for PIKE makes this solution impractical for the TelosB and Tmote Sky platforms, even when considering smaller networks. Alternatively, both ECC and TESLA provide reasonable memory requirements of below 2KB RAM and 3KB ROM which makes them applicable across all platforms. ECC is the most memory-efficient, with an approximatively 50% lower memory footprint when compared to TESLA.

### 5.2.2. Communication overhead

**PIKE/HMAC** – It is intractable to compute precisely the communication overhead during the bootstrapping phase, as it may depend on the relative proximity of the replication nodes, i.e. closer nodes will relay more information to the replication nodes than distant ones. Instead, we evaluate an upper bound as it is dictated by the replication points themselves: all GHT establishment traffic flows through them. Namely, a total of $n/m + (n - n/m) = n$ messages will be sent and received in the worst case, where $n/m$ accounts for the receival and dissemination of information from the *local* $n/m$ nodes, i.e. nodes that are closer to a particular replication point than any other node, and $n - n/m$ denotes the amount of location information concerning the remaining nodes, which is re-routed to the proper replication point. Recall that $n$ is the number of nodes in the network, whereas $m = \lceil \sqrt{n} \rceil$ is the number of replication points in the network. A hash function serves as an index to determine which replication point contains identity/location information about a particular intermediary. The message overhead is given by the bit size $\kappa$ of the identity/location information along with any packet-header overhead $\varrho$. The GHT-overhead is: $C_{\text{PIKE}}^{\text{GHT}} \simeq (\kappa + \varrho)n$.

Path-key establishment consists of a lookup of the intermediary's location followed by a key-exchange between the two peer nodes for which the key is established and their common intermediary. According to PIKE, the communication overhead for a path-key establishment is $\frac{4}{3}\alpha\sqrt{n}$ messages, where $\alpha$ is defined by PIKE as a constant dependent on the range of nodes and shape of deployment area. To provide hop-by-hop authentication, path keys must be established with each of the $N_r R_L$ relaying nodes within. The communication overhead of securing a multi-path is given by the expressions: $C_{\text{PIKE}}^{\text{multipath}} \simeq \frac{4}{3}(K + \varrho)N_r R_L \alpha\sqrt{n}$. Each DATA message carries a HMAC used for end-to-end authentication and a number of $R_L$ HMACs for hop-by-hop authentication. Assuming the size of an HMAC is $h$, the DATA message size overhead incurred is: $C_{\text{PIKE}}^{\text{data}} \simeq hR_L$.

**Digital signatures/ECC** – Communication overhead is incurred during public-key sharing along the sink-to-source and source-to-sink paths. For hop-by-hop authentication the public key of the source needs to be sent along each of the $N_r$ paths and cached by each of the $N_r R_L$ nodes within. The communication overhead to establish a family of multipaths is: $C_{\text{ECC}}^{\text{multipath}} \simeq (K + \varrho)N_r R_L$. The overhead incurred by DATA messages, given an $s$-bits digital signature, is: $C_{\text{ECC}}^{\text{data}} \simeq s$.

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

**Table 11**
Communication overhead summary; $\alpha = .5$, $\beta = 2$, $K = 160$ bits, $\kappa = 48$ bits, $\varrho = 32$ bits, $h = 160$ bits (with SHA-1), $s = 160$ bits. For PIKE-GHT no. of replication points is $m = \lceil \sqrt{n} \rceil$. Number of routes $N_r = 30$.

| Protocol | net. size | Expected route length | Bootstrapping overhead | Operational | |
|---|---|---|---|---|---|
| | $n$ | $R_L$ | | Key-path | Data forwarding |
| | [nodes] | [hops] | [KB] | [KB/multi-path] | [B/packet] |
| PIKE | 1,000 | 24 | 9.77 | 360.00 | 480 |
| | 10,000 | 75 | 97.66 | 3515.63 | 1500 |
| TinyECC | 1,000 | 24 | 0 | 16.88 | 20 |
| | 10,000 | 75 | 0 | 52.73 | 20 |
| TESLA | 1,000 | 24 | 0 | 30.94 | 60 |
| | 10,000 | 75 | 0 | 96.68 | 60 |

**TESLA** – The communication overhead required for initially securing a path is comparable with ECC, since it requires the same public-key mechanism. Additionally for TESLA, however, is the inclusion of a commitment of size $K$ in the message that triggers path-establishment (i.e. S-RREQ for data-forwarding). Accordingly, the communication overhead can be expressed as: $C_{\text{TESLA}}^{\text{multipath}} \simeq (2K + \varrho)N_r R_L$.

During data-forwarding, each data message, in addition to the user-payload, will incorporate the keyed MAC code of the payload, the symmetric key of the previous message and the commitment for the next message, each of which assumed to have size of $K$ bits. In consequence, the DATA-message overhead is: $C_{\text{TESLA}}^{\text{data}} \simeq 3K$.

**Practical comparative analysis**: We have summarized the communication overhead based on the analytical results in Table 11. Since DATA forwarding is expected to dominate the bandwidth usage, it is important to observe that the overhead for providing hop-by-hop authentication with ECC or TESLA is small, namely 20 and 60 additional bytes per data-message respectively, when compared to PIKE. From a feasibility standpoint, the range of 480–1,500 bytes overhead incurred by using PIKE is prohibitive and impractical, even if packet fragmentation is considered, since the MAC802.15.4's packet size is limited to 120 bytes. Since SMP-FPR is expected to execute in large scale sensor networks, long paths are typically the norm, therefore, symmetric key cryptography via PIKE will not scale.

TESLA has a larger message overhead then ECC due to the commitment of the future message and the actual key for the previous message. From a purely communication perspective, ECC seems to be the best solution. Scalability-wise, both ECC and TESLA demonstrate logarithmic performance, as increasing the network size by a factor of 10 increases the associated communication overhead by a factor of 3x for path-establishment.

### 5.2.3. Processing overhead

We leveraged results of existing works, such as TinyECC and TinyHash, to estimate processing costs of the cryptographic techniques we used (generation and verification of the digital signatures or HMAC). We do not include the processing times required to perform lower-network stack operations such as routing and medium access control. We also assume the bootstrapping *processing* times negligible when compared to the security-related overhead. The processing timings that we report for DATA-message forwarding are per-route basis. We denote with $P_g$ the key and digital signatures/HMAC code generation time, assumed comparable, and with $P_v$ the validation time of incoming signatures/codes.

**PIKE/HMAC** – The end-points of a route are the only nodes generating keys and HMACs in SMP-FPR. Thus, in order to provide hop-by-hop authentication, $N_r R_L$ distinct keys need to be generated to be individually shared with relay nodes across the entire family of routes. The multi-path establishment processing overhead is expressed as: $P_{\text{PIKE}}^{\text{multipath}} \simeq N_r R_L P_g$.

For data forwarding, the processing overhead required to successfully transmit a data-packet across an entire path takes into the consideration the generation *and* validation of the HMAC codes, hence: $P_{\text{PIKE}}^{\text{data}} \simeq (P_g + P_v)R_L$, where all on-route validation times are factored in, including the destination validation. The source node will need to generate $R_L$ HMACs for each packet sent.

**Digital signatures/ECC** – ECC needs one single digital signature to provide hop-by-hop authentication data forwarding along $N_r$ routes, hence the processing overhead is reduced to: $P_{\text{ECC}}^{\text{multipath}} \simeq P_g N_r$, and, for each of the routes carrying DATA messages, $P_{\text{ECC}}^{\text{data}} \simeq P_g + R_L P_v$, which accounts for verification overhead at each of the $R_L$ nodes along a path and the key generation at the source.

**TESLA** – Initial path establishment relies on public key cryptography, hence the performance is similar with ECC, accounting for the inclusion of the signed commitment, while path maintenance relies on HMACs. According to the experimental results presented in [7], the computational overhead associated with generation and verification of the commitments is insignificant when compared with the cost of generating an HMAC, a digital signature or performing authentication. Therefore, the processing overhead for securing a family of paths can be approximated as: $P_{\text{TESLA}}^{\text{multipath}} \simeq N_r P_{g_{(\text{ECC})}}$, where the subscript indicates that the generation times are dictated by ECC execution. When it comes to data-forwarding along a path, the processing overhead is dominated by HMAC generation timing at the source and one verification of the code at the sink and at each $R_L - 1$ relay nodes. Therefore, the data-forwarding processing overhead along an entire path is: $P_{\text{TESLA}}^{\text{data}} \simeq P_{g_{(\text{HMAC})}} + R_L P_{v_{(\text{HMAC})}}$.

**Practical comparative analysis**: We report the processing times for TelosB platform, which is commonly[1] analyzed in both TinyECC and TinyHASH. Based on the results in [31],

[1] We have used the results corresponding to Tmote Sky from TinyECC as representative for TelosB, since both platforms share the same MSP430 processor clocked at the same 8 Mhz frequency.

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

**Table 12**
Processing time overhead summary; $\alpha = .5$, $\beta = 2$, $K = 160$ bits, $\kappa = 48$ bits, $\varrho = 32$ bits, TelosB and Tmote Sky motes. For PIKE-GHT no. of replication points is $m = \lceil \sqrt{n} \rceil$. Number of routes $N_r = 30$.

| Protocol | Net. Size $n$ | Expected Route length $R_L$ | TinyECC Generation time $P_g$ | TinyECC Validation time $P_v$ | TinyHASH Generation time $P_g$ | TinyHASH Validation time $P_v$ | Operational Secure path | Operational Data Forwarding |
|---|---|---|---|---|---|---|---|---|
| | [nodes] | [hops] | [s] | [s] | [s] | [s] | [s] | [s] |
| PIKE | 1,000 | 24 | – | – | 0.11 | 0.11 | 75.60 | 5.04 |
| | 10,000 | 75 | – | – | 0.11 | 0.11 | 236.25 | 15.75 |
| TinyECC | 1,000 | 24 | 21.00 | 43.00 | – | – | 43.00 | 1,053.00 |
| (w/o opt) | 10,000 | 75 | 21.00 | 43.00 | – | – | 43.00 | 3,246.00 |
| TinyECC | 1,000 | 24 | 1.58 | 2.02 | – | – | 2.02 | 50.06 |
| (w/ opt) | 10,000 | 75 | 1.58 | 2.02 | – | – | 2.02 | 153.08 |
| TESLA | 1,000 | 24 | 1.58 | 2.02 | 0.11 | 0.11 | 2.02 | 2.63 |
| | 10,000 | 75 | 0.58 | 2.02 | 0.11 | 0.11 | 2.02 | 7.98 |

for example, the execution time for HMAC+SHA1 algorithms, on TelosB motes, is approximatively $P_g \simeq P_v = 105$ ms for both HMAC generation and verification. Table 12 shows the processing timings for TelosB motes and serves as a comparative reference for the expected overhead differential. These results reaffirm, however, the main drawback of using exclusively public key cryptography, as in TinyECC: prohibitive processing timings, which can induce very long delays especially in data-forwarding. Clearly, DS/ECC cannot be a feasible solution for hop-by-hop authentication since traversing a path can take minutes (on the slower TelosB platform considered, at least), even with all optimizations enabled. For example, traversing a 24-hop route will take approximatively 1 min.

By comparison, PIKE/HMAC and TESLA induce far lower data delivery latencies, albeit the path establishment time in PIKE/HMAC in the orders of minutes is prohibitive. TESLA has low forwarding latencies via HMAC mechanism, and low setup latencies via optimized TinyECC. For example, securing a path takes only 2 additional seconds, on par with public key cryptography performance (ECC optimized) and two orders of magnitude faster than PIKE/HMAC, while the data delivery latencies are nearly half of the best values of PIKE, conversely, it can support data streams of double data rates. We remark however that we have solely considered the best performances achievable via optimized TinyECC, since the memory overhead required to implement these optimizations are well within the admissible memory bounds of real platforms and likely to be implemented as such.

### 5.2.4. Latency overhead

All the processing and communication overhead introduce non-negligible latencies during the bootstrapping, multi-path establishment and data forwarding.

**PIKE/HMAC** – The typical duration of the bootstrapping phase is increased due to the GHT service underneath PIKE. The exact latency increase is difficult to compute analytically due to queuing and other MAC-layer protocol specific overheads (i.e. beacons, sleep schedules, etc.). Assuming quasi-parallel GHT setup, we can devise a lower bound on GHT's setup time, which is dictated by the communication overhead induced through one replication point, that is, $2C^{\text{GHT}}_{\text{PIKE}}/R$, accounting for both transmissions and receptions, where $R$ denotes the data-rate of a particular mote platform. In the case of multi-path establishment, the latency overhead is $T^{\text{multipath}}_{\text{PIKE}} \simeq C^{\text{multipath}}_{\text{PIKE}}/R$. We note that multi-path establishment latency overhead represents an upper bound and assumes that paths are sequentially built; this is a reasonable assumption if MAC contention is to be avoided, since the communication overhead for path-establishment is significant in PIKE/HMAC. Same analysis extends to the latency incurred for data forwarding along a single route, that is $2C^{\text{data}}_{\text{PIKE}}/R + P^{\text{data}}_{\text{PIKE}}$, which include transmission and receiving timings in addition to authentication processing overhead.

**Digital signatures/ECC** – ECC/digital signatures have no bootstrapping overhead. In the case of secure multi-path establishment and data-forwarding, since there is no significant on-path communication overhead when using public keys, paths may be built in parallel, hence the overhead is reduced to $2C^{\text{multipath}}_{\text{ECC}}/(N_r R) + P^{\text{multipath}}_{\text{ECC}}/N_r$, while for data forwarding along a single route is $2C^{\text{data}}_{\text{ECC}}/R + P^{\text{data}}_{\text{ECC}}$.

**TESLA** – Performance expressions are similar to DS/ECC, i.e. for securing a family of routes the latency incurred is given by $2C^{\text{multipath}}_{\text{TESLA}}/(N_r R) + P^{\text{multipath}}_{\text{TESLA}}/N_r$, while for data forwarding along a single route is $2C^{\text{data}}_{\text{TESLA}}/R + P^{\text{data}}_{\text{TESLA}}$.

**Practical comparative analysis**: Table 13 illustrates the calculated latency values expected to be exhibited on TelosB platforms, as an comparative example. Correspondingly, the latency due to PIKE's initial GHT establishment ranges between 640 ms for small network sizes and high-data rate radios of 250 kbps, up to 6.4*s* for large network sizes.

In the case of securing paths, there is a significant trade-off that can be achieved between path-establishment latency and data-delivery latency. For example, using the fully optimized version of TinyECC allows for a quick 2-s multi-path establishment, however, the data-delivery latency becomes very large, i.e. up to 4 min for nodes comprised of 10,000 nodes, severely limiting the data rate of the user data stream. Alternatively, PIKE's setup time is the order of minutes, however it achieves better data rate margins. If we denote with $x$ the number of multi-paths that can be used simultaneously for data delivery, the maximum data rate achievable is $0.2x$ packets per second for smaller-sized networks and $0.0625x$ for large networks, with PIKE.

Table 13 illustrates that, if using TESLA, one can expect very good performance during both path establishment and data forwarding phases. For example, TESLA doubles the maximum supported throughput when compared to PIKE, while, at the same time, achieves the best path-establishment timings.

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

**Table 13**
Associated latency overhead for TelosB platforms $\alpha = 0.5$, $\beta = 2$,$R = 250$ kbps For PIKE-GHT no. of replication points is $m = \lceil \sqrt{n} \rceil$. Number of routes $N_r = 30$.

| Protocol | Net. size | Expected route length | Bootstrapping overhead | Operational | |
|---|---|---|---|---|---|
| | $n$ [nodes] | $R_L$ [hops] | [s] | Secure path [s/multipath] | Data forwarding [s/packet/path] |
| PIKE | 1,000 | 24 | 0.64 | 76.39 | 5.78 |
| | 10,000 | 75 | 6.40 | 243.93 | 22.95 |
| TinyECC | 1,000 | 24 | 0 | 43.04 | 1,053.03 |
| (w/o opt) | 10,000 | 75 | 0 | 43.12 | 3,246.10 |
| TinyECC | 1,000 | 24 | 0 | 2.06 | 50.09 |
| (w/ opt) | 10,000 | 75 | 0 | 2.14 | 153.18 |
| TESLA | 1,000 | 24 | 0 | 2.09 | 2.72 |
| | 10,000 | 75 | 0 | 2.23 | 8.27 |

**Table 14**
Practical energy overhead with TelosB node ($R = 250$ kbps). For PIKE-GHT no. of replication points is $m = \lceil \sqrt{n} \rceil$. Number of routes $N_r = 30$.

| Protocol | Net. size | Expected route length | Bootstrapping overhead | Operational | |
|---|---|---|---|---|---|
| | $n$ [nodes] | $R_L$ [hops] | [mJ/node] | Secure path [mJ/source/multipath] | Data forwarding [mJ/relay/packet] |
| PIKE | 1,000 | 24 | 48.00 | 440.09 | 3.44 |
| | 10,000 | 75 | 480.00 | 1586.79 | 8.33 |
| TinyECC | 1,000 | 24 | 0.00 | 233.69 | 237.02 |
| (w/o opt) | 10,000 | 75 | 0.00 | 236.87 | 233.81 |
| TinyECC | 1,000 | 24 | 0.00 | 12.40 | 11.36 |
| (w/ opt) | 10,000 | 75 | 0.00 | 15.57 | 11.12 |
| TESLA | 1,000 | 24 | 0.00 | 13.65 | 0.88 |
| | 10,000 | 75 | 0.00 | 19.46 | 0.86 |

After presenting the individual analysis of the energy overheads for each of the mechanisms in Section 5.1, we conclude this section with a comparative analysis of the energy overheads among PIKE, TinyECC and TESLA in Table 14, pointing out to an additional aspect to consider: the cost. For example, considering a small 75 mAh battery that can provide energy for a couple of days under moderate operation, the cost of securing a family of routes under PIKE with security level for networks of 10,000 nodes is approximatively 0.15% of the total battery capacity of the source node. Assuming no limits on the data rates, the cost overhead of relaying a stream of data with a sampling interval of 5 s for 10 h is 11% using optimized TinyECC for the source node and 0.33% for a relay node. In comparison, TESLA achieves 97% energy savings when compared with PIKE for path establishment under 1000 nodes networks, and 99% under larger networks. However, when compared to DS/ECC, the energy overhead in TESLA is 10% and 25% respectively for path establishment, sensibly more costly. Fortunately, path-establishment is a relatively infrequent operation and its cost can be rapidly amortized during the data-forwarding phase, where TESLA actually achieves energy savings of 92% regardless of the network size when compared to DS/ECC, and between 74% and 90% when compared to PIKE. This is an additional justification for our selection of TESLA, since it is the most cost effective solution by a significant margin.

## 6. Defense against attacks via selective forwarding

In the previous two sections, we presented a detailed analysis justifying our choice of *TESLA* [7] as a mechanism for authentication and integrity. We now proceed with presenting the details of three mechanisms of SMP-FPR: k-EF, k-RPEF, and path diversity monitoring scheme (PDMS). The solutions identified are also applicable for selective *delaying* of messages in the original MP-FPR. Specifically, k-EF provides resilience against delaying DATA messages, k-RPEF addresses the delays of QUERY, ACK and UPDATE messages, and PDMS addresses the delays of RREQ messages.

Before continuing with the details of each mechanism, in order to better position their role in the overall SMP-FPR, we summarize again each of the possible attacks in MP-FPRand show the applicability of the mechanisms to a particular attack in Table 15. For completeness, we have retained the categories which are handled by *TESLA*.

### 6.1. Approaches to Defend Against Selective Forwarding

*Proactive mechanisms* typically relay replicas of the messages along multiple paths. For example, *k-redundant depender graphs* [32] provide every node with $k$ disjoint paths towards a sink, thus guaranteeing delivery even when $k - 1$ paths in between have failed. The *k-RIP* [33] represents an improvement by providing probabilistic redundant forwarding to $k$ randomly picked neighboring nodes, with the main advantage of decreasing the vulnerability to route discovery – e.g., Sybil attacks. Other methods rely on a deterministic finite path-diversity model to increase robustness by a priori discovering a family of multi-path routes [34–37], to provide redundancy between two end-points [38].

*Reactive mechanisms* employ detection and isolation techniques of misbehaving nodes. One approach abstracts the

**Table 15**
Effectiveness of attacks and defense mechanisms.

| Attack | | Defense | | | |
|---|---|---|---|---|---|
| Type | Actions/messages | TESLA | k-EF | k-RPEF | PDMS |
| PD | Alter $C_e$ in QUERY/UPDATE | √ | – | – | – |
| PDD | Modify $N_r$ in QUERY | √ | – | – | – |
| PDD | Modify $r_i$ in RREQ/ACK | √ | – | – | – |
| PDD | Modify $L_{src}$ in RREQ/ACK | √ | – | – | – |
| PDD | Modify $t_s$ in RREQ | √ | – | – | – |
| PDD | Drop/delay RREQ | – | – | – | √ |
| PDD | Drop ACK | – | – | √ | – |
| FPI | Drop/delay UPDATE | – | – | √ | – |
| FPI | Modify $L_{src}$ in UPDATE | √ | – | – | – |
| WP | Modify $C_e$ in RREQ | √ | – | – | – |
| FLP | Modify $r_i$ in RREQ/DATA | √ | – | – | – |
| DoS | Drop QUERY | – | – | √ | – |
| DoS | Drop DATA | – | √ | – | – |
| DoS | Drop ACK | – | – | √ | – |
| DoS | Delay DATA | – | √ | – | – |
| DoS | Modify $r_i$ in DATA | √ | – | – | – |
| DoS | Modify $r_i$ in ACK | √ | – | – | – |
| DP | Modify payload in DATA | √ | – | – | – |
| DSI | Modify $L_{src}$ in QUERY | √ | – | – | – |

adversarial activity as a link-quality deterioration and addresses the problem from a robustness perspective. For example, ODSSBR [39], avoids the under-performing links by using a modified version of a secure route discovery protocol that uses a link-quality metric. Similarly, [40] uses a weight-management scheme to quantify link-quality. The net effect of these schemes is avoidance of the compromised areas, allowing for a graceful degradation of service. In contrast, other schemes adopt a radical detection and isolation model: nodes exhibiting unexpected behavior are removed immediately and permanently from the network's topology. Approaches consist of: (1) performing end-to-end monitoring and statistical analysis of traffic patterns – the *pathrater* technique [41], and (2) exploiting topological properties in sensor networks, i.e. multiple nodes are within collision domain, which enables overhearing of node's communication for the purpose of detecting unexpected communication patterns [42–45].

### 6.2. Our approach

Recall that MP-FPR uses five type of messages sent via two forwarding mechanisms, EF and SGP. Selective forwarding attacks against these messages are shown in Table 2.

**DATA messages**: As they are sent via EF forwarding where multi-path is readily available, resilient forwarding can be provided by sending them along subsets of such EF routes. We refer to this mechanism as *k-EF*, where *k* is the *degree of replication* and denotes the number of copies sent along distinct routes.

**QUERY, UPDATE and ACK messages**: Since they rely on SGP forwarding, no routes are readily available. There are two possibilities for providing *k*-resilience in this case: (1) replacement of the standard SGP mechanism with a k-shortest path routing [46] (which we refer to as k-SGP), and (2) adapt MP-FPR protocol to rely directly on the field-based forwarding provided by EF to forward copies along multiple *on-the-fly* built routes, which we will refer to as k-RPEF (Reverse Path Electrostatic Forwarding). We adopt k-RPEF for the following reasons: (1) it is relatively easy to implement since it relies on the same forwarding mechanism as in EF, (2) it simplifies the network-protocol stack by removing SGP altogether, and (3) its redundant paths inherit the non-braiding property of field-based routing, which cannot be guaranteed with k-SGP.

**RREQ messages**: They cannot benefit from a redundancy mechanism, since they are bound to the single route they probe and implicitly construct. We propose path diversity monitoring scheme (PDMS), a scheme that reactively attempts to compensate for any deficiencies in path diversity by persistently building more routes until the path diversity quota is met.

### 6.3. k-EF defense mechanism

The k-EF mechanism provides replication of DATA messages using the set of active routes resulting from the route establishment phase. The degree of replication is given by the value of $k \leq N_r$, where $N_r$ represents the maximum number of routes that can be established. We use a random selection scheme to select $k$ paths from the total of $N_r$ possible. Recall that routes are uniquely identified by the route index $r_i$ – equivalently, the tangent angle of a particular field-line exiting the source. Hence, in a *k*-redundant scheme, the indexes are randomly selected from the $\varphi_{N_r}$ set, without replacement.

### 6.4. k-RPEF defense mechanism

k-RPEF provides redundant forwarding of QUERY, UPDATE and ACK messages towards the source nodes. Forwarding is still based on electrostatic field lines, but traverse in opposite direction of the field vectors, towards the source. To enable this, we reverse the algebraic sign of the charge's magnitudes corresponding to the sink and specific source charge – for reverse path forwarding only. For example, if a sink and a source have charges of $Q_{src} = -1 \times 10^{-19}$ Coulombs and $Q_{snk} = +1 \times 10^{-19}$ Coulombs respectively, k-RPEF's field lines will be built on the set of charges $Q_{src} = +1 \times 10^{-19}$ Coulombs and $Q_{snk} = -1 \times 10^{-19}$ Coulombs instead. The only charge that changes is the one of the sources to which we intend to forward the message – whereas the charges of other sources remain unchanged. This is required to prevent messages from reaching other source nodes by hopping on their field lines. In addition, the sign reversal is performed in isolation from other sources, i.e. such information is not broadcasted and it is only used locally. This will cause the field line vectors to point towards the chosen source node rather than the sink, guiding the associated routes without further modification of the forwarding algorithm.

### 6.5. Path diversity monitoring scheme (PDMS)

Since RREQ messages are uniquely associated with the routes they are forwarded through, their replicas cannot follow different routes, rendering the k-RPEF mechanism inadequate. PDMS enables the source node to persistently probe for new routes until the user-specified *path diversity quota* of distinct routes $N_r$ is achieved. The crux of PDMS is bypassing compromise nodes via sequence of attempts.

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

PDMS cannot be used as a standalone solution for path diversity deflation attacks carried out via ACK messages, for the following reason. Recall that, in the absence of k-RPEF mechanism, ACK messages are sent via SGP forwarding – thus, compromising the single reverse path will block the acknowledgment phase and regardless of the number of attempted routes, they will never get acknowledged. PDMS, however, can provide *compensatory* benefits if the k-RPEF resilient mechanism is already employed for ACK messages, and our experimental results will demonstrate this benefit.

SMP-FPR attempts to maintain the generation of evenly distributed routes in the physical field. Thus, a *sequence* of routes will be enabled and probed in a manner that takes into consideration the existing distribution of routes and attempts to fill any existing "gaps" in order to provide a graceful degradation in the presence of attacks.

PDMS is a multi-phase process. The first phase performs the same functions as in the original MP-FPR protocol: for a given source node $sn_s$, a *sequence* $S_{s,1}$ of $N_r$ evenly distributed routes is generated $S_{s,1} = \langle \phi_i | \phi_i = \frac{2\pi}{N_r} i, i \in \{1, \ldots, N_r\} \rangle$ – and subsequently, the routes are probed. If the path diversity quota is not met, the next route construction phase is initiated. Let $A_{s,j}$ denote the ordered (by the tangent angle with respect to the source-to-sink line) sequence of all the *active* acknowledged routes up to phase $j > 1$ (inclusive). If the path diversity quota is not being met, i.e. $|A_{s,j}| < |N_r|$, a subsequent phase $j+1$ is initiated. In each subsequent phase, a new distinct *base routing sequence* $S_{s,j+1}$ is being generated such that $|S_{s,j+1}| = N_r$. The base routing sequence at phase $j > 1$ is generated by incrementing the tangent angles used in the previous phase by a fixed angle $\delta$ in a counter-clockwise direction. As opposed to the very first phase, however, the order in which the routes from $S_{s,j+1}$ will be probed for RREQ will changed. This is motivated by the desideratum to fill in gaps from the previous iteration(s) via routes from the current one. In addition, the probing process can be interrupted whenever the path diversity quota is achieved. To prevent wasteful energy resources under severe adversarial conditions, one may also limit the number of phases that can be executed to some predefined value $K \geq 2$.

For a given limit of probing phases $K$, in the worst case scenario, the union of all base routing sequences is $\bigcup_{j=1}^{j=K} S_j = \langle r_i | r_i = \frac{2\pi}{K \cdot N_r} i, i \in 1, \ldots, N_r \rangle$, hence a total of $N_p = K \cdot N_r$ distinct and evenly distributed routes may be probed by PDMS. An example for $K = 3$ and $N_r = 8$ routes per phase, with rotation increment $\delta = 15^\circ$ is shown in Fig. 3.

To promote spatially even distribution, when resequencing the routes from $S_{s,j+1}$, priority should be given to routes situated in the vicinity of a route whose failure to acknowledge the RREQ in the previous phase(s) has contributed to a "gap".

The advantage of the proposed PDMS scheme versus one in which field lines for routes are randomly selected is twofold: (1) PDMS maintains control of the probed routes and can (attempt to) target areas with lower densities of routes – i.e., in *immediate* vicinity of failed routes); and (2) it promotes the separation among the new routes by guaranteeing a minimum path-spacing $\delta$, to reduce the possibility of route merging effects.

The prioritization mechanism for selecting the (RREQ probing sequence) of the routes generated in phase $j+1$ is based on the angular-gaps between any two adjacent routes from the sequence $A_j$ of active ones up to phase $j$, together with the candidate ones generated from the base routing sequence $S_{s,j+1}$. We utilize the sequence $G_{j+1}$ of $\langle$*(gap_value, route_valuerangle)* pairs, sorted in a descending order of the *gap_value* attribute. In case two pairs have the same *gap_value*, they are sorted in ascending order of the *route_value* attribute (which is unique). The details of this mechanism are formalized in Algorithm 1 where, for clarity,

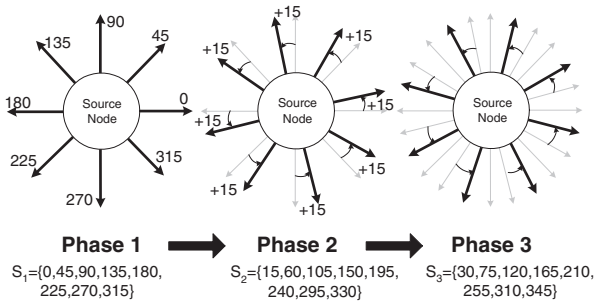



**Fig. 3.** Base routing sequences for $K = 3$ route construction phases and $N_r = 8$ routes per phase. Each phase's routes' indexes are obtained by applying a rotational shift of routes' indexes in previous phase by $\delta = 2\pi/K \cdot N_r = 15^\circ$.

**Algorithm 1** Priority base route generation in a phase of PDMS.

**Input:**
$j$: current PDMS phase number ($j = 1$ for 1*st* construction phase generation)
$K$: maximum number of phases // assume $j < K$
$A_{j-1}$: set of acknowledged RREQ routes after phase $j-1$ ($A_j = \emptyset$ if $j = 1$)
$N_r$: targeted number of routes
$\delta = 2\pi/(K \cdot N_r)$: phase-increment
**Output:**
$\overline{S}_j$ - **sequence** of routes for $j+1$, to be used for RREQ.

1: $S_j = \langle r_i | r_i = \frac{2\pi}{N_r} i + \delta(j-1), i \in 1, \ldots, N_r \rangle$
2: $\overline{S}_j = S_j \cup A_{j-1}$
3: $B = \langle \overline{S}_j[last], \overline{S}_j \rangle$ // Wrap around sequence
4: $first = 0$
5: $last = Max(0, |B| - 1)$;
6: $G_j = \langle \emptyset \rangle$ // sequence of $\langle$ (gap_value,route_index) $\rangle$
7: **for** $i = first; i \leq last; i = i + 1$ **do**
8: $gap\_value = (B[i+1] - B[i])\%2\pi$
9: $gap\_value = B[i]$
// Insert new (gap_value,route_value) entry, preserving the ordering
10: $G_j \leftarrow G_j \cup \langle gap\_value, route\_value \rangle$
11: **end for**
12: **for** $i = first; i \leq last; i = i + 1$ **do**
13: $\overline{S}_j[i] \leftarrow G_j[i].route\_value$ // Re-arrange $\overline{S}_j$ into a prioritized sequence
//of routes to be used in the RREQs generations for Phase 2.
14: **end for**
15: $\overline{S}_j \leftarrow \overline{S}_j \setminus A_{j-1}$ // Eliminate the already-acknowledged routes from phase $j-1$

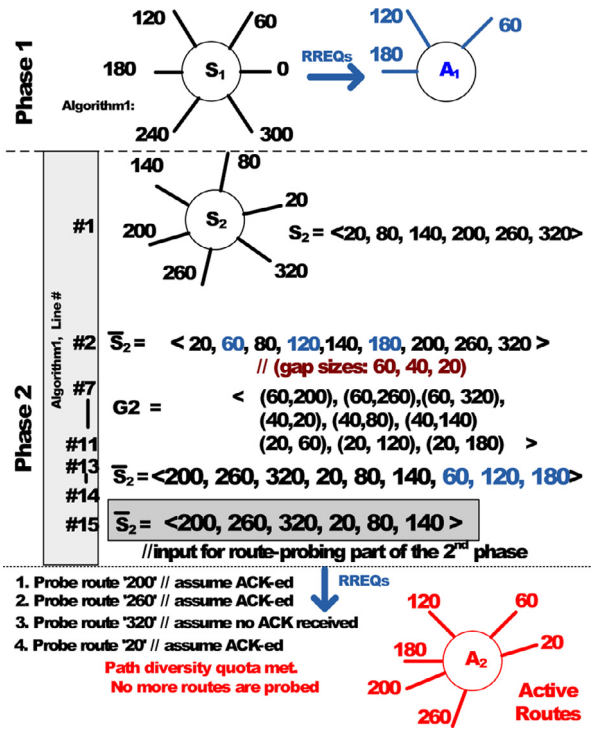



**Fig. 4.** Priority base route generation example with $K = 3$, $N_r = 6$, $\delta = 20^\circ$ (route indexes in degrees): Phase 1 completed, with 3 ACK-ed routes, $A_1 = \{60^\circ, 120^\circ, 180^\circ\}$. Algorithm 1 is applied and a new prioritized sequence of route indexes to be probed is generated: $\overline{S}_2 = \langle 200^\circ, 260^\circ, 320^\circ, 20^\circ, 80^\circ, 140^\circ \rangle$. After executing the RREQs, the sequence of routes becomes: $A_2 = \{20^\circ, 60^\circ, 120^\circ, 180^\circ, 200^\circ, 260^\circ\}$ of cardinality 6 ($= N_r$), which meets the path diversity quota and phase 2 is interrupted.

we omit the subscript of the source node $sn_s$. For two sorted sequences $A$ and $B$, we use $A \cup B$ to denote the sorted sequence obtained by merging them.

An example of the priority-based generation of routes in PDMS is presented in Fig. 4. The key observation is the reordering of the basic route sequences, based on the gaps from the previous phases (lines 7–11 of Algorithm 1).

## 7. Experimental evaluation

In this section we evaluate the effectiveness of the proposed defense mechanisms and demonstrate their viability. After an overview of the experimental settings and metrics used, we present the overhead analysis of the TESLA integrity mechanisms (our solution of choice – cf. Section 5.2), followed by detailed experimental reports for the selective-forwarding resilience mechanisms, i.e. k-RPEF, PDMS and k-EF.

### 7.1. Simulation settings and metrics

We use SIDnet-SWANS [47,48], an open-source large scale sensor network simulator built upon the scalable architecture of JiST-SWANS [49], which in turn is based on a high-performance JiST engine. It carries an adapted version of ns-2's MAC802.15.4 protocol and same signal propagation models.

**Network configuration**: The simulated environment consists of a set of 750 homogeneous nodes configured with: (1) 20 kbps transmission/reception rate, (2) MAC802.15.4 protocol, (3) 5 s idle-to-sleep interval (i.e., nodes that are not actively involved in routing enter a low energy consumption state after 5 s of continuous idling), and (4) power consumption characteristics based on Mica2 Motes specifications [16]. The nodes were placed in an area of 1000 × 1000 ft. To reduce the simulation time while preserving the validity of the observations, nodes used a small battery with an initial capacity of 35 mAh, for a projected lifespan of several tens of hours under moderate load.

**Application settings.** The tested scenario consists of four distinct, long-term, continuous, point-to-point queries rooted at a common sink node centrally located within the network. The four corresponding source nodes are evenly distributed around the sink node within the corner regions of a grid-based partitioning of the network. This configuration has two advantages: (1) it provides approximatively 90% spatial coverage of the relay area to the network resources (nodes) and (2) it creates a context of four adjacent families of routes, which enables investigating of the family path intersection attacks via selective forwarding of UPDATE-messages violating the disjointness property of *different* source-sink families of routes. In addition, the four queries are injected in the network *sequentially* at 10 min simulated time intervals. The path diversity quota has been set to $N_r = 30$ routes, and the PDMS's path offset $\delta = 4^\circ$ for a maximum of $N_p = 90$ pool of candidate routes. Each experiment captures 8 h of simulated time. Data transmission interval of the point-to-point queries to the designated sink is 4 s. We increased the set of attacking nodes, which are randomly and uniformly selected, ranging from 5% to 30% of the total number of sensors in the network.

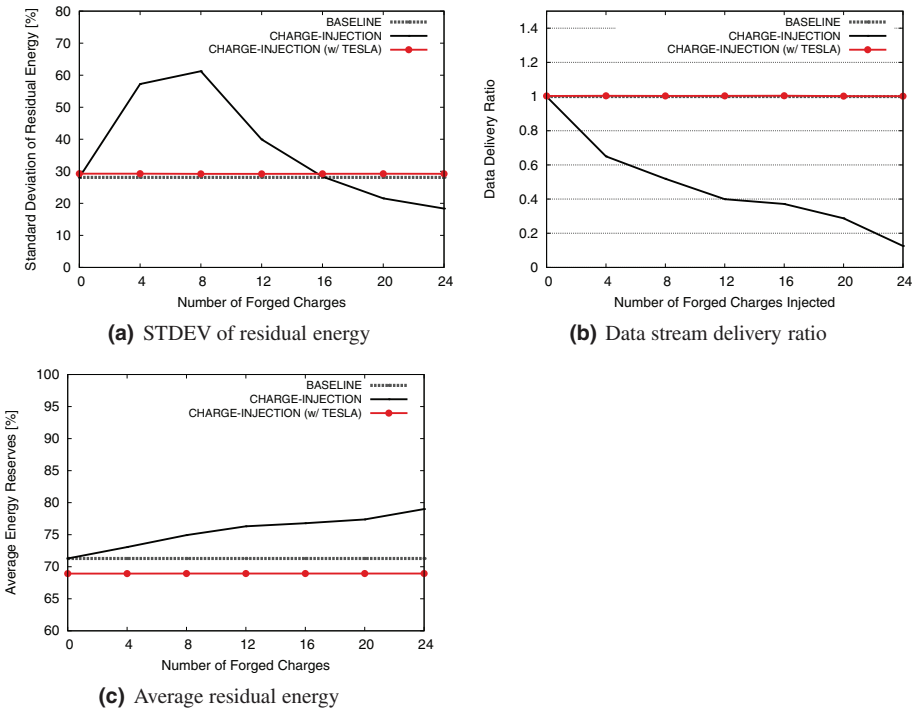
**Metrics**: We monitor and measure:

(1) *Successful query dissemination rate* – the ratio between the number of queries received at the corresponding source(s) vs. the total number of queries submitted through the sink node.

(2) *Average residual energy levels* $\overline{E}$ in the entire network, normalized relative to the capacity of a fully charged battery $E_{\max}$. The effectiveness of the workload balancing paradigm and its associated energy consumption distribution is measured via the standard deviation of the percentage-representation of the residual energy reserves $E_\sigma$. If $E_i(t) \le E_{\max}$ denotes the residual energy level of a sensor node $sn_i$ at time-instant $t$, then the average energy level in a network of $N$ nodes is $\overline{E}(t) = \frac{1}{N}\sum_{i=1}^{N} E_i(t)/E_{\max}$, and $E_\sigma$ is computed as:

$$E_\sigma(t) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(E_i(t) - \frac{1}{N}\sum_{j=1}^{N}E_j(t))^2}. \quad (1)$$

(3) *Packet-delivery ratio* $\eta = n_{\text{rcv}}/n_{\text{exp}}$, established between the number of packets actually received $n_{\text{rcv}}$ by the sink node and the total number of packets sent $n_{\text{exp}}$ by the source node and expected at the sink over an interval of time. In multipath settings, the delivery ratio accounts for the successful transmission of one (of the possible many) copies




**(a)** STDEV of residual energy

**(b)** Data stream delivery ratio

**(c)** Average residual energy

**Fig. 5.** Path deflection attack with and without Tesla.

of a packet. The depreciation of the packet deliver latency is also monitored for our overhead analysis.

### 7.2. Evaluation of TESLA for integrity and authentication

To demonstrate the effectiveness of TESLA used for integrity and authentication in SMP-FPR, we mounted a path deflection attack by altering electrostatic charge information via either QUERY or UPDATE messages. Path deflection is the most representative attack since: (1) it is an attack that targets unique characteristics of MP-FPR, (2) it requires very little resources to mount and (3) it can yield most damaging effects over the energy consumption patterns.

**Energy balancing and data delivery rate**: We create a path deflection attack by generating forged-charges randomly placed in various areas of the network through the UPDATE messages. We vary the number of forged charges between 4 and 24, the upper bound value being enough to create major loss of connectivity in the network, as the experiments will show.

Fig. 5(a) shows the impact of inserting invalid charge information on (disruption of) the energy balancing. As can be seen, SMP-FPR is very sensitive to this type of attack: even a small number of forged charges, for example 4, are enough to drastically affect the uniformity of the energy consumption, as the standard deviation of residual energy reserves nearly doubles. This is because of the severe path deflection and agglomeration of routes in narrow physical areas, due to the repulsive effect of multiple forged charges. In these conditions, most of the alternate paths within a family merge and converge towards a single path type of routing in the relay area, effectively degrading the original MP-FPR towards a single-path.

With a larger number of forged charges, i.e. $> 8$, there exists an apparent improvement of the energy-balance, as shown in Fig. 5(a). This is an extreme side effect of charge forgery attack: user perceived data DoS. Namely, it is possible that field lines are deflected enough so that *all* of the associated routes are too long to be accepted in the route construction phase. The net result is a complete isolation between affected sources and the sink. This lack of connecting routes prevents the data-stream from being sent to the sink, resulting in energy-savings by *not performing* the required workload. To demonstrate that this is the case, we capture the impact over the data-delivery rate in Fig. 5(b). As illustrated, data-delivery rate drops because of this effect, improving the network wide average of residual energy levels by up to 12% (cf. Fig. 5(c)), when 24 forged charges are randomly injected in the network. Fig. 5(a) also shows that when TESLA is enabled as a defense mechanism the performance of the system with respect to residual energy is similar with the performance when no attack takes place.

**TESLA energy overhead evaluation**: Fig. 5(a)–(c) demonstrate that TESLA has a small energy-overhead, independent of the dimension of the attack. It can be observed that TESLA's impact over the energy-balancing mechanism is below 3%, whereas the impact over the network-wide average residual energy levels is maintained below 5% (cf. Fig. 5(c)).

The overall latency of the data stream delivery is increased with TESLA. However, this is due to the key-generation process that takes place at the source node prior to message transmission, as well as on-route key-verification process. Fig. 6 demonstrates that the TESLA mechanism

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

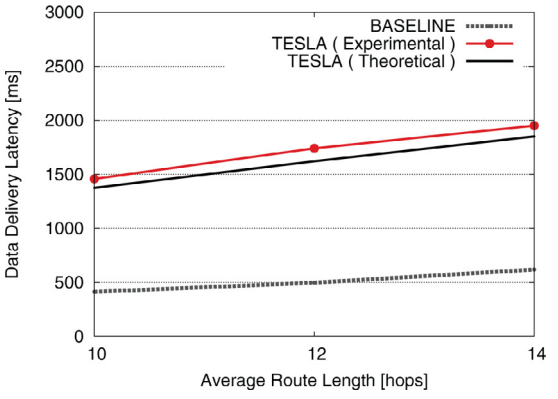



**Fig. 6.** No attack: End-to-end data delivery latency with TESLA.

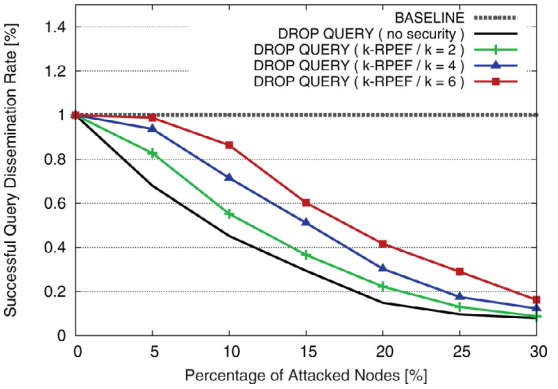



**Fig. 7.** Selective forwarding of QUERY: Query dissemination rate with and without k-RPEF.

increases the end-to-end data delivery latency by a factor of three – a 1–1.5 s latency increase over the unsecured MP-FPR alternative for routes with 10–14 hops in length respectively. Average route lengths of 10, 12 and 14 hops were achieved from networks of 750, 1000 and 1250 nodes respectively, varying the size of the deployment area.

The result presented in Fig. 6 confirms that the latency overhead increases linearly with the path length, as the analysis in Section 4 indicated. Fig. 6 includes the theoretical end-to-end delivery latencies based on the results of Table 13 for the path-length considered. We note that the experimental results indicate approximately 5% additional latency overhead vs. the theoretical expectations. This is due to several realistic factors that are taken into consideration during simulation, such as transmission delays due to contention in wireless medium – phenomenon that is more pronounced near the sink, where all the routes converge.

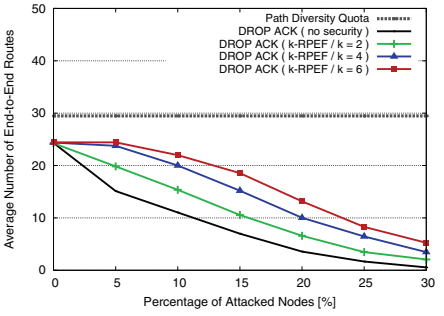
### 7.3. Effectiveness of k-RPEF against selective forwarding

The k-RPEF mechanism provides redundant forwarding of QUERY, ACK, and UPDATE messages towards the source nodes.

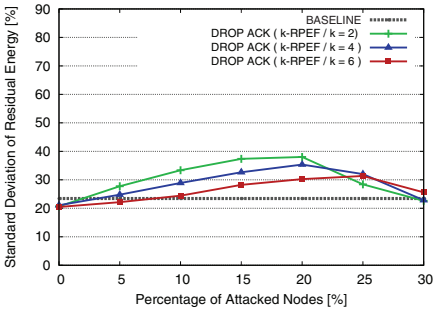
**Selective forwarding of QUERY messages**: Attacks carried during the query dissemination phase target QUERY messages en-route to the source nodes their processing. Fig. 7 shows that targeting the QUERY messages represents an easy and effective way to block query processing capabilities in the network. For example, by targeting 5% of the sensor nodes, an attacker can expect to impact 30% of the queries submitted. To demonstrate the effectiveness of the k-RPEF replication mechanism, we tested settings with degrees of replication of $k = 2, 4$ and 6. For example, when 6 replicas of QUERY messages are sent, SMP-FPR proves to become nearly insensitive to the same number of attacks on QUERY messages (5%) – fewer than 1% query disseminations fail. Overall, we note an approximate reduction of successful attacks by 5% for every additional path used for replication, slightly lower under very intense attack settings of more than 25% compromised nodes. This information is relevant for deciding the number of replicas and multi-paths a query message will be sent along, when specific security needs and risk factors are known. The number of k-RPEF multi-paths can be increased solely based on the security requirement, as the impact on the energy reserves is negligible.

**Selective forwarding of ACK messages**: Dropping ACK messages leads to an outcome similar to the one caused by dropping RREQ messages, as comparing Fig. 8(a) with Fig. 10(a) demonstrates. Namely, with only 5% of the nodes compromised, the effective number of routes has been reduced by nearly 50%, slightly worse than the selective forwarding of RREQ messages.

ACK messages are different from RREQ messages – they are not tightly coupled to a particular field line to be forwarded along, hence replicas can be created and forwarded along distinct paths. Fig. 8(a) demonstrates a significant




**(a)** Path diversity




**(b)** STDEV of residual energy

**Fig. 8.** Selective forwarding of ACK messages with and without k-RPEF.

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

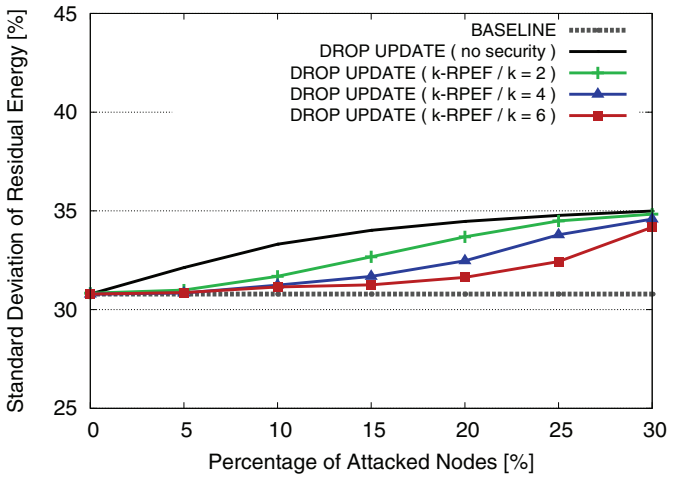



**Fig. 9.** Selective forwarding of UPDATE: STDEV of residual energy with and without k-RPEF.

improvement provided by the k-RPEF mechanism, ranging from approximatively 30% when the degree of replication is $k = 2$, to nearly 100% as the degree of replication is increased to $k = 6$. We can also observe a linear dependency of the improvement to the number of replicas, each additional replica providing a benefit of 15%, on average.

The impact of the degree of replication $k$ is illustrated in Fig. 8(b). Larger values of $k$ promote larger set of routes that improve energy consumption balancing at a rate of approximatively 8% for each additional replica, consistent for attacks comprised of less than 20% nodes. When the attacking base is increased beyond the 20% mark, the energy balance has "apparent improvement" – similar to the one discussed for RREQ messages.

Because the original MP-FPR protocol sends ACK messages via the SGP mechanism, i.e. along a unique path, it has a higher risk of loosing end-to-end connectivity if the attacks target ACK messages. For example, a single compromised node along the SGP route will compromise the entire route and consequently the entire acknowledgment phase. In extreme: (1) either no ACK message is lost and end-to-end connectivity is achieved with unaffected families of routes, or (2) *all* ACK messages are being dropped and no routes are established. In both cases, energy is maintained balanced: (1) due to lack of effective workload and (2) due to diverse families of routes. When it comes to the unprotected MP-FPR alternative, the energy imbalance in SMP-FPR "improved" monotonically as the adversarial activity amplified, due to increased likelihood of end-to-end connectivity loss, which is why we omit its inclusion in Fig. 8(b).

**Selective forwarding of UPDATE messages**: A critical information carried by UPDATE messages is the charge value based on which the non-braiding property of the electrostatic field lines is maintained. Dropping UPDATE messages undermines this property, leading to family path intersection attacks, where increased and uneven energy consumption manifests in the areas were paths pertaining to distinct families of routes start braiding. The effect is more pronounced under high data rate streams where temporary queuing and risk of wireless contention are higher. For this type of experiment we have increased the data delivery transmission rate from 0.25 messages per second to 1 message per second, at each source node. Fig. 9 illustrates the impact of the wild-path condition attack on the residual energy balance. As shown, attacks carried during route establishment phase may yield up to 15% degradation of energy consumption balance for the data-rate considered.

Fig. 9 also shows that using k-RPEF effectively alleviates the family path intersection attacks. Namely, when the degree of replication is set to $k = 6$, the degradation of energy balance is maintained below 2% for bases of attacks that cover up to 15% of the nodes, and below 5% when 20% of nodes are compromised.

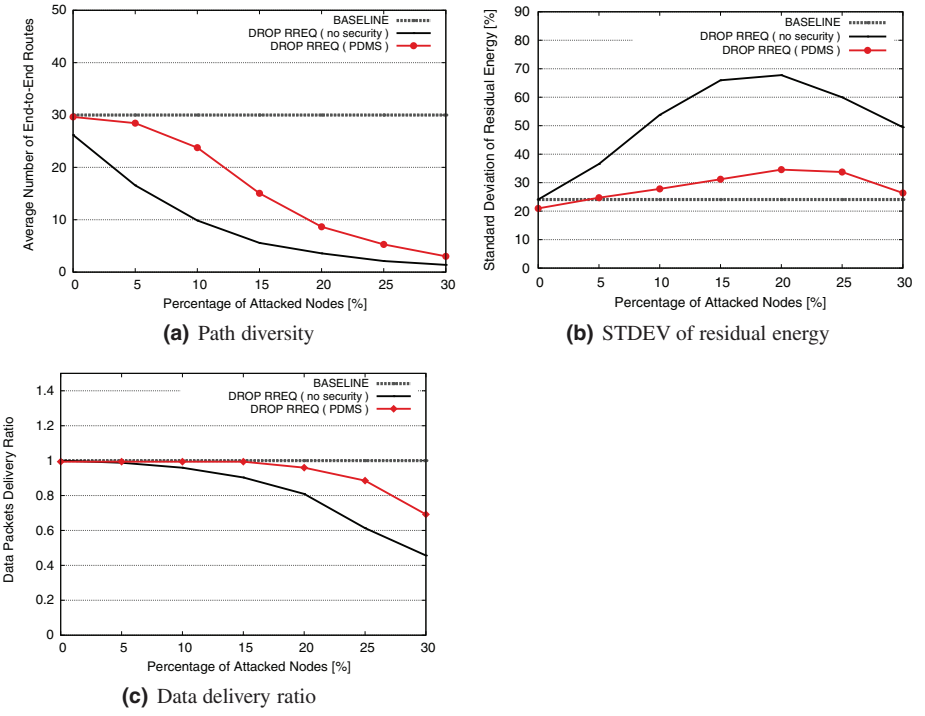
**Sensitivity to degree of replication of k-RPEF**: Under long-term queries, the amount of traffic generated by QUERY, ACK and UPDATE messages, and the associated bandwidth and energy costs, are insignificant when compared to the large-volume DATA streams. We have, however, demonstrated the incremental benefits of expanding the number of message replicas and sending them along distinct paths (cf. Figs. 7, 8(a), 8(b), 9). It can be observed that when the base of attack is reduced, i.e. up to 20% of compromised nodes, increasing the degree of replication $k$ still provides overall benefits.

It can also be consistently observed that the benefit of increasing the degree of replication when the base of attacking nodes is larger than 20% diminishes. This is a consequence of a large decrease in connectivity for a particular message-flow, due to larger density of compromised nodes in the relay area – situation in which detection/isolation mechanisms are additionally required.

### 7.4. Effectiveness of PDMS against selective forwarding

PDMS provides protection against selective forwarding for REQ messages. While it cannot be used as a standalone solution for path diversity deflation attacks on ACK messages, it has a compensatory effect when k-RPEF is enabled.

**Selective forwarding of RREQ messages**: We have simulated path-diversity deflation attacks via selective




**(a)** Path diversity

**(b)** STDEV of residual energy

**(c)** Data delivery ratio

**Fig. 10.** Selective forwarding of RREQ messages with and without PDMS.

forwarding of RREQ messages. We note that these experimental results are also representatives of alternative attack mechanism – significantly delaying RREQ messages, whereby paths exhibiting high latencies are not acknowledged. Both mechanisms have an identical adversarial outcome: reduced set of routes, which PDMS will compensate for.

Fig. 10(a) shows the high sensitivity to path diversity deflation attacks, as even with 5% compromised nodes, the number of paths is reduced by 40%, compared to the non-adversarial settings. PDMS is part of the SMP-FPR which significantly improves the resilience to route establishment attacks, as for the same base of attacking nodes, the reduction of alternative paths is only 6%. Consequently, the attacker needs to consider tripling the attacking base, i.e. compromising approximatively 15% nodes, to achieve the same damaging effect as in the unprotected MP-FPR. From a different perspective, under the same adversarial conditions, PDMS scheme enables achievement of up to 140% richer families of routes as compared to unprotected MP-FPR.

Fig. 10(a) illustrates an additional benefit of PDMS: improving path diversity even under non-adversarial conditions. Namely, even when there are no compromised nodes, SMP-FPR yields an average of 17% fewer routes than the user-specified quota ($N_r = 30$ in these settings). This is because SMP-FPR, just like the original MP-FPR, discards routes that do not meet the end-to-end latency requirements (cf. Section 2), such as overly long paths. PDMS implicitly addresses this issue by persisting in probing routes until the path diversity quota is being met, since it is oblivious of the underlying reasons why certain routes are not acknowledged. Hence, PDMS provides additional (potential) improvement to the original, unprotected MP-FPR. The benefit can also be observed in Fig. 10(b), showing that PDMS scheme achieves a 12% improvement in terms of energy balancing over MP-FPR in non-adversarial environments.

Attacks carried during route establishment cause reduction of the effectiveness of the workload balancing. Fig. 10(b) illustrates the depreciation of energy-balancing as the number of compromised nodes is increased – a 110% increase in standard deviation of the residual energy levels when only 10% of the nodes are compromised. PDMS helps maintaining even energy consumption distribution, achieving below 15% depreciation under the same settings. The workload imbalance tops with 175% depreciation when 20% nodes maliciously drop RREQ messages, and "recover" as the number of attacks is further increased. We recall that the apparent recovery is due to the loss of end-to-end connectivity. When absolutely no routes can be established between the source and sink nodes due to too many compromised nodes, the data stream is virtually absent and the afferent messages are dropped at the source. Energy savings are being achieved in the relay-area due to the lack of the data stream workload. To demonstrate that this is the case, we analyze in sequel the impact of attacks carried via selective forwarding of RREQ messages over the data delivery ratio.

As shown in Fig 10(c), the sensitivity to message-dropping of RREQ messages is significantly reduced when compared to the reduction in path diversity under the same settings. Namely, when 5% of nodes are compromised, the impact to message dropping is below 1%. This is because the reduction of path-diversity does not affect message delivery, but the total absence of connecting routes does. As it can be observed,

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

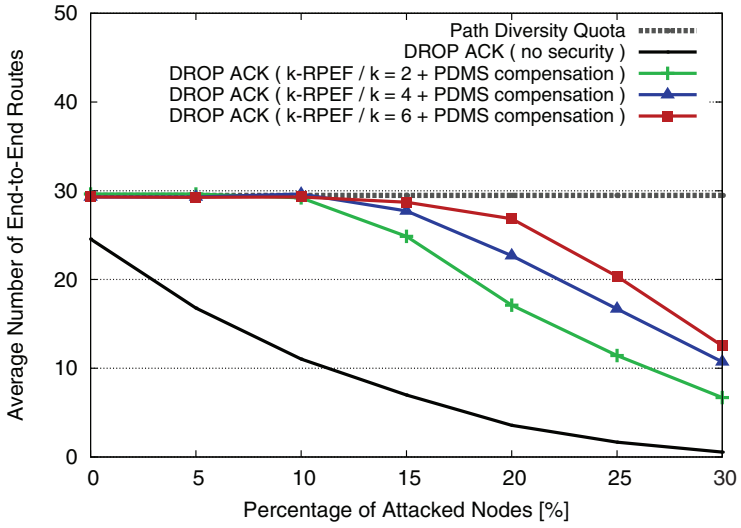



**Fig. 11.** Selective forwarding of ACK: Path diversity with and without k-RPEF and PDMS mix.

when the base of attacks is increased to 30% nodes, the average number of disconnected source-to-sink topologies is around 50%. The PDMS enables higher data-message delivery ratios since the family of routes it yields is consistently larger and the risk of non-connectivity is consequently lowered. PDMS forces an attacker to consider a much larger base of attacking nodes, an average of 20% more, to render SMP-FPR ineffective in achieving end-to-end connectivity as with the unprotected MP-FPR, with respect to the data stream deliverability.

**Compensatory effect of PDMS to k-RPEF during attacks via selective forwarding of ACK messages**: Both k-RPEF and PDMS mechanisms provide protection against path diversity deflation under adversarial conditions. However, they are fundamentally different: k-RPEF is a *proactive* mechanism – attempting to reduce the risk of failing to ACK a route; whereas PDMS is *reactive* – attempting to build a new routes upon failure of a previously attempted route. Since dropping either of ACK and RREQ messages leads to a route construction failure, PDMS will compensate for both in an attempt to meet the path diversity quota – i.t., it will react to dropping of ACK messages as well. We have analyzed k-RPEF and PDMS solutions in isolation, and now we present an experimental analysis where both of these methods are combined.

Fig. 11 shows the improvement in path diversity when PDMS and k-RPEF are used together. This combination provides a rather strong defense against selective forwarding of ACK messages when the base of compromised nodes is below 10% – the path diversity remains unaffected. Moreover, the PDMS component enables SMP-FPR to reach the path diversity quota even under this adversarial scenario. It takes 30% of the nodes to be compromised, in order to achieve comparative protection of k-RPEF running in isolation against 20% of compromised nodes. From the perspective of sheer resilience to adversarial activity, PDMS improves the performance of k-RPEF, on average, by 90%.

Note that PDMS, in isolation, cannot provide any benefit against selective forwarding of ACK messages, since SGP mechanism is employed for relaying ACK messages in the original MP-FPR. That is, if the SGP established sink-to-source path is compromised, *all* ACK messages will be dropped, including those acknowledging routes that PDMS attempts to build as replacement. In other words, compromising the unique route in SGP mechanism effectively nullifies the PDMS's benefits with respect to selective forwarding of ACK messages.

Energy balancing also benefits by enabling the PDMS to operate in conjunction with the k-RPEF solution. As Fig. 12 demonstrates, considering a degree of replication of $k = 6$ and an attacking base of 20%, the disruption of energy balancing is of only 16%, i.e. a nearly 50% improvement when compared to the equivalent performance of running k-RPEF in isolation (cf. previous results in Fig. 8(b)).

### 7.5. Effectiveness of k-EF against selective forwarding

We now study the impact of data DoS carried via selective forwarding of DATA messages and the benefits of applying a multipath strategy via k-EF mechanism. It involves using subsets of acknowledged routes, rather than on-demand paths as in k-RPEF. Due to the high-volume of data traffic, replication of such traffic must be limited in order to avoid: (1) wasting resources and (2) bandwidth saturation, especially considering the proximity of the sink node where data flows converge. We have tested scenarios with degree of replication of $k = 2, 3$ and 4 only.

**Selective forwarding of DATA messages**: Fig. 13(a) illustrates the consequence of increasing the number of attacking nodes that target DATA messages: a 45% degradation in DATA packet delivery with a only a small base of 5% nodes, and nearly 90% degradation when the number of compromised nodes is increased to 15%. This vulnerability is particularly important as the user-payload within dropped DATA messages cannot be recovered. Adopting a multipath approach proves to be beneficial in this situation as well: at the minimum, the effect is reduced by a factor of two, i.e. from 45% to

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

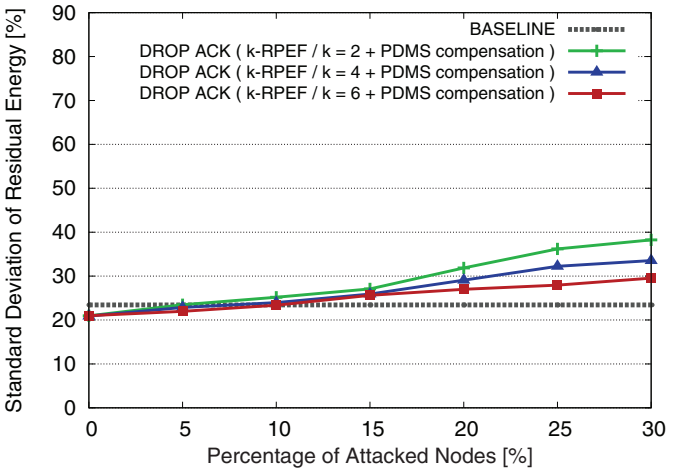
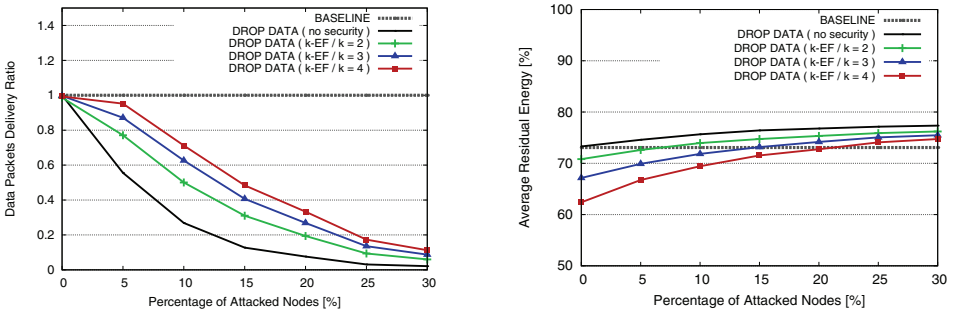



**Fig. 12.** Selective forwarding of ACK: STDEV of residual energy with and without k-RPEF and PDMS mix




**(a)** Data delivery ratio **(b)** Average residual energy

**Fig. 13.** Selective forwarding of DATA messages with and without k-EF.

23% message drops when only 2 replication paths are used, and less than 2% when 4 replication paths are used, considering 5% compromised nodes. This relative improvement in SMP-FPR is consistent regardless of the number of compromised nodes. From an attacker standpoint, achieving the same net effect as unprotected MP-FPR nearly doubles for, e.g., 4 replication paths.

The cost of providing resilience against data DoS via selective forwarding of DATA messages is reflected in increased energy consumption. Assuming a no-attack environment, Fig. 13(b) shows an overhead varying between 3% and 15% as the number of multipaths is increased from $k = 2$ to $k = 4$ – likely to increase further with higher transmission rates. The number of compromised nodes does not appear to have a direct negative impact on the energy consumption. It is, however, the case that energy savings are achieved when DATA messages are being dropped along a path due to an undesirable reduction of the workload. As shown in Fig. 13(b), the residual energy reserves increase monotonically with the reduction of the successful delivery of data messages from Fig. 13(a).

**Sensitivity to the degree of replication of k-EF**: Relaying a large DATA stream from source nodes towards a sink node induces energy and bandwidth costs that cannot be ignored. Fig. 13(a) and (b) represent the benefit, respectively the cost, of increasing the degree of replication. A detailed cost-benefit analysis is beyond the scope of this work however, our experiments provide some guidelines.

As an example, if the application comprise correlation analysis or outlier detection in which completeness of the data stream has high priority, under reduced security risk scenarios, the system may be configured to use a higher degree of replication. For example, if $k = 4$, i.e. 4 distinct paths are employed to relay copies of a given DATA message, under 5% compromised nodes settings, it is expected a success rate of data-stream delivery of 98% (cf. Fig. 13(a)). However, under the same settings, the maximum time-span for information delivery is projected to be reduced by 15%, considering DATA messages transmission rate of .25 messages per second. The projection is based on a corresponding reduction of the average residual energy reserves (cf. Fig. 13(b)), expectedly lower under increasing data rates. Overall, each increment of the degree of replication has approximately 5% improvement of successful DATA stream delivery, at a cost of 1% energy consumption under the DATA message transmission rates considered.

## 8. Related work

Recent work on the security of sensor networks has focused on proposing key management schemes that can be used to bootstrap other services [25–27,50,51], addressing

general attacks such as Sybil [52] and replication [53] attacks, as well as identifying basic attacks in wireless sensor networks [54].

The security of geographical routing protocols using physical nodes' locations was studied in [55,56] for sensor networks and in [13,57] for ad-hoc networks. Most of the works focus on preventing malicious modifications of the destination location in packets, verifying neighbor location information, and preventing message dropping. Another main area of work in securing geographic routing is the protection of the location service, which includes [11,12].

Security of a gradient based routing for sensor networks, has been studied in [58]. This work distinguishes from ours as follows: (1) the work surveys a generic list of attacks and countermeasures that do not focus on the specifics of the potential-field routing, while we address specific risks introduced by the MP-FPR protocol in all phases of the protocol operation and analyze these risk factors through extensive experimental analysis; (2) although potential-field routing and electrostatic field-based routing are both instances of the gradient based routing, their implementation is fundamentally different: the former is a *stateful* protocol, where routes are established based on distance metrics obtained by means of hop-counting, while SMP-FPR does not maintain routing information and relies only on the distribution of discrete charge information for forwarding purposes; (3) field-based routing has been proposed initially in the context of large scale, dense mesh networks and there is no focus on energy consumption and workload distribution, whereas SMP-FPR generalizes the usability of gradient based routing to arbitrary distributions with possible low densities of nodes and focuses on the energy aspect.

Often, applications' contexts demand that a WSN is organized in hierarchical routing structures or divided in multiple clusters – posing different security challenges. For instance, SecLeach [59] protocol addressed the security aspects of hierarchical routing protocols with dynamic clusters (re)formation [60], coupling random key pre-distribution and $\mu$TESLA. Privacy may also be an additional requirement for position-based routing, for example preserving the privacy of sink location [61] or the location of nodes in general [62,63]. However, these settings are, in a sense, orthogonal to the ones considered in this paper.

Geographic routing remains an active area of research due to intrinsic benefits of exploiting location for routing. A survey of geography-based single-path routing can be found in [64], whereas an approach that considers the challenges of large scale sensor networks is presented in [65]. A more comprehensive survey on the topic is presented in [66]. Energy-aware geographical routing was studied in [67]. Other works have also recognized the benefits of using multipath routing in large-scale sensor networks for improving workload balancing and delivery robustness. Trajectory-based forwarding, which rely on multiple non-braided paths via parametric curves for single source and sink scenarios, have been presented in [68,69]. A natural extension to multiple sink, multiple-path is challenging because route disjointness cannot be easily guaranteed when adopting parametric trajectory models, therefore field, potential and gravity-based routing methodologies, which exploit physical phenomena properties to facilitate the creation of non-braiding paths, have been recently investigated [70–73]. Despite the interest in gradient based routing, little work has been done to address the security of such protocols.

## 9. Conclusions

We analyzed the feasibility of providing security services to various attacks targeting different phases of the multi-pole field persistent routing (MP-FPR) – an instance of the electrostatic field based routing for sensor networks. Towards that, we proposed the secure variant – SMP-FPR and we provided two-fold augmentations. First, we investigated several cryptographic mechanisms for integrity and authentication primitives, considering public (TinyECC) and symmetric (PIKE) key, and hybrid (TESLA) cryptographic approaches. Subsequently, we proposed three solutions k-EF, k-RPEF and PDMS that exploit the native multi-path nature of MP-FPR, in order to improve resilience to selective data forwarding attacks.

Given the importance of energy consumption in WSNs and SMP-FPR's goals in these aspects, we focused on changes to the energy-consumption patterns induced by the security primitives. We have experimentally demonstrated that SMP-FPR energy provisions can be significantly affected under an adversarial environment, however, effective security solutions that exploit MP-FPR's multi-path routing model can be implemented with minimal overhead.

## Acknowledgments

This research was supported by the NSF grants CNS 0910952 and III 1213038, and ONR grant N00014-14-10215.

## References

[1] I.F. Akyildiz, M.C. Vuran, Wireless Sensor Networks, Wiley, 2010.
[2] C. Lemmon, S.M. Lui, I. Lee, Geographic forwarding and routing for ad-hoc wireless network: a survey, in: NCM, 2009, pp. 188–195.
[3] D. Niculescu, B. Nath, Trajectory based forwarding and its applications, in: MOBICOM, 2003, pp. 260–272.
[4] N.T. Nguyen, A. Wang, P. Reiher, G. Kuenning, Electric-field-based routing: a reliable framework for routing in MANETs, SIGMOBILE Mobile Comput. Commun. Rev. 8 (2) (2004) 35–49, doi:10.1145/997122.997129. http://portal.acm.org/citation.cfm?id=997122.997129
[5] G. Trajcevski, O.C. Ghica, P. Scheuermann, M. Zuniga, R. Schubotz, M. Hauswirth, Improving the energy balance of field-based routing in wireless sensor networks, in: IEEE Globecom, 2010, pp. 1–5.
[6] O. Ghica, G. Trajcevski, M. Zuniga, P. Scheuermann, R. Schubotz, M. Hauswirth, Multi-pole field persistent routing with bounded delay in wireless sensor networks, Technical Report, Dept. of EECS, Northwestern University, 2011. http://www.eecs.northwestern.edu/docs/techreports/2011_TR/NWU-EECS-11-04.pdf
[7] A. Perrig, R. Canetti, J.D. Tygar, D.X. Song, Efficient authentication and signing of multicast streams over lossy channels, in: IEEE Symposium on Security and Privacy, 2000, pp. 56–73.
[8] O.C. Ghica, C. Nita-Rotaru, G. Trajcevski, P. Scheuermann, Security of electrostatic field persistent routing: attacks and defense mechanisms, in: European Dependable Computing Conference, 2012, pp. 102–113. http://doi.ieeecomputersociety.org/10.1109/EDCC.2012.23.
[9] R. Fonseca, S. Ratnasamy, J. Zhao, C.T. Ee, D. Culler, S. Shenker, I. Stoica, Beacon vector routing: scalable point-to-point routing in wireless sensornets, in: NSDI, 2005, pp. 329–342.
[10] T. He, C. Huang, B.M. Blum, J.A. Stankovic, T.F. Abdelzaher, Range-free localization schemes for large scale sensor networks, in: MOBICOM, 2003, pp. 81–95.
[11] J. Dong, K.E. Ackermann, B. Bavar, C. Nita-Rotaru, Mitigating attacks against virtual coordinate based routing in wireless sensor networks, in: WiSec, 2008, pp. 89–99. http://doi.acm.org/10.1145/1352533.1352548.

[12] X. Wu, C. Nita-Rotaru, On the security of distributed position services, in: SecureComm, 2005, pp. 35–46.
[13] J.-H. Song, V.W.S. Wong, V.C.M. Leung, Secure position-based routing protocol for mobile ad hoc networks., Ad Hoc Netw. 5 (1) (2007) 76–86.
[14] S. Ganeriwal, C. Pöpper, S. Capkun, M.B. Srivastava, Secure time synchronization in sensor networks, ACM TISSEC 11 (4) (2008) 23:1–23:35.
[15] J. Barnickel, U. Meyer, Secsywise: a secure time synchronization scheme in wireless sensor networks, in: ICUMT, 2009, pp. 1–8.
[16] Crossbow products, http://www.xbow.com/.
[17] Tmote sky: reliable low-power wireless sensor networking eases development and deployment, http://www.moteiv.com/products-tmotesky.php.
[18] A. Liu, P. Ning, Tinyecc: a configurable library for elliptic curve cryptography in wireless sensor networks, in: ISPN, 2008, pp. 245–256.
[19] W.T. Zhu, Y. Xiang, Argus: a light-weighted secure localization scheme for sensor networks, in: ATC, 2009, pp. 164–178.
[20] W. Ammar, A. ElDawy, M. Youssef, Secure localization in wireless sensor networks: a survey, CoRR abs/1004.3164 (2010).
[21] C. Harsch, A. Festag, P. Papadimitratos, Secure position-based routing for vanets, in: VTC Fall, 2007, pp. 26–30.
[22] H. Krawczyk, M. Bellare, R. Canetti, Hmac: keyed-hashing for message authentication, RFC 2104 (1997) 1–12.
[23] N. Koblitz, Elliptic curve cryptosystems, Math. Comput. 48 (1987) 203–209.
[24] D. Johnson, A. Menezes, S.A. Vanstone, The elliptic curve digital signature algorithm (ecdsa), Int. J. Inform. Secur. 1 (1) (2001) 36–63.
[25] L. Eschenauer, V.D. Gligor, A key-management scheme for distributed sensor networks, in: ACM CCS, ACM Press, 2002, pp. 41–47.
[26] H. Chan, A. Perrig, D.X. Song, Random key predistribution schemes for sensor networks, in: IEEE Symposium on Security and Privacy, 2003, pp. 197–213.
[27] H. Chan, Pike: peer intermediaries for key establishment in sensor networks, in: Proceedings of IEEE INFOCOM, 2005, pp. 524–535.
[28] A. Perrig, R. Szewczyk, V. Wen, D. Culler, J.D. Tygar, Spins: security protocols for sensor networks, in: Wireless Networks, 2001, pp. 189–199.
[29] S.P. Miller, B.C. Neuman, J.I. Schiller, J.H. Saltzer, Kerberos authentication and authorization system, in: Project Athena Technical Plan, 1988.
[30] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, S. Shenker, Ght: a geographic hash table for data-centric storage, in: WSNA, 2002, pp. 78–87.
[31] H. Lee, Y. Choi, H. Kim, Implementation of TinyHash based on hash algorithm for sensor network, in: Proc. of World Academy of Science, Engineering, and Technology, 2005, pp. 135–139.
[32] R. Wright, P.D. Lincoln, J.K. Millen, Efficient fault-tolerant certificate revocation, in: In ACM Conference on Computer and Communications Security, ACM CCS, 2000.
[33] S. Zhu, C. Yao, D. Liu, S. Setia, S. Jajodia, Efficient security mechanisms for overlay multicast-based content distribution, in: ACNS, 2006, pp. 40–55.
[34] K. Sanzgiri, B. Dahill, B.N. Levine, C. Shields, E.M. Belding-Royer, A secure routing protocol for ad hoc networks, in: ICNP, 2002, pp. 78–89.
[35] S. ju Lee, Split multipath routing with maximally disjoint paths, in: in Ad hoc Networks, in Proc. of IEEE ICC, 2001.
[36] A. Koul, R.B. Patel, V.K. Bhat, Double split based secure multipath routing in adhoc networks, in: ARTCom, 2009, pp. 835–839.
[37] B. Awerbuch, R. Curtmola, D. Holmer, H. Rubens, C. Nita-Rotaru, On the survivability of routing protocols in ad hoc wireless networks, in: IEEE SECURECOMM, IEEE Computer Society Press, 2005, pp. 327–338.
[38] P. Papadimitratos, Z.J. Haas, Secure message transmission in mobile ad hoc networks, Ad Hoc Netw. 1 (1) (2003) 193–209.
[39] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru, H. Rubens, Odsbr: an on-demand secure byzantine resilient routing protocol for wireless ad hoc networks, ACM TISSEC 10 (4) (2008).
[40] H. Tan, On mitigating malicious behavior against routing in wireless networks, in: WCNC, 2007.
[41] S. Marti, T.J. Giuli, K. Lai, M. Baker, Mitigating routing misbehavior in mobile ad hoc networks, in: MOBICOM, 2000, pp. 255–265.
[42] A. Nasipuri, R. Castañeda, S.R. Das, Performance of multipath routing for on-demand protocols in mobile ad hoc networks, MONET 6 (4) (2001) 339–349.
[43] A.A. Pirzada, C. McDonald, Establishing trust in pure ad-hoc networks, in: ACSC, 2004, pp. 47–54.
[44] Y.-A. Huang, W. Lee, A cooperative intrusion detection system for ad hoc networks, in: SASN, 2003, pp. 135–147.
[45] I. Khalil, S. Bagchi, C. Nita-Rotaru, N.B. Shroff, Unmask: utilizing neighbor monitoring for attack mitigation in multihop wireless sensor networks, Ad Hoc Netw. 8 (2) (2010) 148–164.
[46] D. Ganesan, R. Govindan, S. Shenker, D. Estrin, Highly-resilient, energy-efficient multipath routing in wireless sensor networks., Mobile Comput. Commun. Rev. 5 (4) (2001) 11–25.
[47] O. Ghica, G. Trajcevski, P. Scheuermann, Z. Bischoff, N. Valtchanov, Sidnet-swans: a simulator and integrated development platform for sensor networks applications, in: SenSys, 2008, pp. 385–386.
[48] Sidnet-swans, http://www.eecs.northwestern.edu/~ocg474/SIDnet.html.
[49] Jist-swans, http://jist.ece.cornell.edu/index.html.
[50] W. Du, J. Deng, Y.S. Han, P.K. Varshney, J. Katz, A. Khalili, A pairwise key predistribution scheme for wireless sensor networks, TISSEC 8 (2) (2005) 228–258.
[51] D. Liu, P. Ning, R. Li, Establishing pairwise keys in distributed sensor networks, TISSEC 8 (1) (2005) 41–77.
[52] J. Newsome, E. Shi, D. Song, A. Perrig, The sybil attack in sensor networks: analysis and defenses, in: IPSN, 2004, pp. 259–268. http://doi.acm.org/10.1145/984622.984660.
[53] B. Parno, A. Perrig, V. Gligor, Distributed detection of node replication attacks in sensor networks, in: 2005 IEEE Symposium on Security and Privacy, 2005, pp. 49–63, doi:10.1109/SP.2005.8.
[54] C. Karlof, D. Wagner, Secure routing in wireless sensor networks: attacks and countermeasures, in: WSNA, 2003, pp. 113–127.URL citeseer.ist.psu.edu/article/karlof02secure.html
[55] N. Abu-Ghazaleh, K.-D. Kang, K. Liu, Towards resilient geographic routing in wsns, in: Q2SWinet, 2005, pp. 71–78.
[56] T. Zahariadis, P. Trakadas, H. Leligou, S. Maniatis, P. Karkazis, A novel trust-aware geographical routing scheme for wireless sensor networks, Wireless Person. Commun. 69 (2) (2013) 805–826, doi:10.1007/s11277-012-0613-7.
[57] T. Leinmüller, C. Maihöfer, E. Schoch, F. Kargl, Improved security in geographic ad hoc routing through autonomous position verification, in: VANET, 2006, pp. 57–66.
[58] D. Sharma, Security of field based routing, Student Thesis SA-2008-08 (2008).
[59] L.B. Oliveira, A.C. Ferreira, M.A. Vilaça, H.C. Wong, M.W. Bern, R. Dahab, A.A.F. Loureiro, Secleach - on the security of clustered sensor networks, Signal Process. 87 (12) (2007) 2882–2895.
[60] W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: HICSS, 2000, pp. 8020–8029.
[61] B. Ying, D. Makrakis, H. Mouftah, A protocol for sink location privacy protection in wireless sensor networks, in: GLOBECOM, 2011, pp. 1–5, doi:10.1109/GLOCOM.2011.6133922.
[62] K. El Defrawy, G. Tsudik, Alarm: anonymous location-aided routing in suspicious manets, IEEE Trans Mobile Comput. 10 (9) (2011a) 1345–1358. http://doi.ieeecomputersociety.org/10.1109/TMC.2010.256.
[63] K. El˜Defrawy, G. Tsudik, Privacy-preserving location-based on-demand routing in manets, Select. Areas Commun IEEE J. 29 (10) (2011b) 1926–1934, doi:10.1109/JSAC.2011.111203.
[64] S. Ruehrup, Theory and practice of geographic routing, in: Ad Hoc and Sensor Wireless Networks: Architectures, Algorithms and Protocols, Bentham Science, 2009.
[65] A.-M. Kermarrec, G. Tan, Greedy geographic routing in large-scale sensor networks: a minimum network decomposition approach, in: MobiHoc, 2010, pp. 161–170.
[66] F. Cadger, K. Curran, J. Santos, S. Moffett, A survey of geographical routing in wireless ad-hoc networks, Commun. Surv. Tutorials, IEEE 15 (2) (2013) 621–653, doi:10.1109/SURV.2012.062612.00109.
[67] H. Huang, G. Hu, F. Yu, Energy-aware geographic routing in wireless sensor networks with anchor nodes, Int. J. Commun. Syst. 26 (1) (2013) 100–113, doi:10.1002/dac.1335. http://dx.doi.org/10.1002/dac.1335
[68] M. Desai, N. Maxemchuk, Polar coordinate routing for multiple paths in wireless sensor networks, in: WOWCOM, 2010, pp. 1–9.
[69] O. Ghica, G. Trajcevski, P. Scheuermann, N. Valtchanov, Z. Bischof, Controlled multi-path routing in sensor networks using Bezier curves, Comput. J. 54 (2) (2011) 230–254.
[70] C. Wu, R. Yuan, H. Zhou, A novel load balanced and lifetime maximization routing protocol in wireless sensor networks, in: VTC Spring, 2008, pp. 113–117.
[71] J. Li, S. Ji, H. Jin, Q. Ren, Routing in multi-sink sensor networks based on gravitational field, in: ICESS, 2008, pp. 368–375. http://dx.doi.org/10.1109/ICESS.2008.14.
[72] S. Rührup, H. Kalosha, A. Nayak, I. Stojmenovic, Message-efficient beaconless georouting with guaranteed delivery in wireless sensor, ad hoc, and actuator networks, IEEE/ACM Trans. Netw. 18 (1) (2010) 95–108.
[73] X. Li, J. Yang, A. Nayak, I. Stojmenovic, Localized geographic routing to a mobile sink with guaranteed delivery in sensor networks, Select. Areas Commun. IEEE J. 30 (9) (2012) 1719–1729, doi:10.1109/JSAC.2012.121016.

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016

**Oliviu C. Ghica** received his B.Sc. degree in Computer and Electrical Engineering from "Politehnica" University of Bucharest, Romania, in 2002, followed by a M.Sc. and Ph.D. degrees from the Department of Electrical Engineering and Computer Science at Northwestern University, USA, in 2006 and 2011, respectivelly. His research interest span through spatio-temporal data management and routing in large-scale wireless sensor networks, with a particular interest in energy efficiency and lifetime benefits of algorithmic implementations. He was an active player in open-source communities focusing on simulation methodologies for sensor network applications. Presently, he is presently affiliated with Acrom System Integrations.

**Cristina Nita-Rotaru** is an Associate Professor in the department of Computer Science at Purdue University. She leads the Dependable and Secure Distributed Systems Laboratory. She received BS and MS degrees from Politechnica University of Bucharest, Romania, in 1995 and 1996, and a Ph.D. degree in Computer Science from Johns Hopkins University in 2003. She served on the technical program committee of over 40 conference in networking, distributed systems, and security. She received the NSF CAREER award. She served as an Associate Editor for ACM Transactions on Information Security and she is currently an Associate Editor for IEEE Transactions on Dependable and Secure Computing and IEEE Transactions on Mobile Computing. Her research interests include security and fault-tolerance for distributed systems and networks.

**Goce Trajcevski** received his B.Sc. degree from the University of Sts. Kiril i Metodij, and his MS and Ph.D. degrees from the Department of Computer Science at the University of Illinois at Chicago. His main research interests are in the areas of spatio-temoral data management, routing and data management in wireless sensor networks, and reactive behavior in dynamic systems. He has published over 90 papers in refereed conferences and journals and received a Best Paper Award at the CoopIS conference (2000), Best Paper Award at the IEEE MDM conference (2010) and Best Short Paper Award at ACM MSWiM conference (2013). His research has been funded by BEA, Northrop Grumman Corp., NSF and ONR. He has served as an associate editor at ACM DiSC, and is presently an associate editor of GeoInformatica and ACM Transactions on Spatial Algorithms and Systems (TSAS). He has served on program and organizing committees in numerous conferences and workshops. Currently, he is an Assistant Chairman with the Department of Electrical Engineering and Computer Science at the Northwestern University.

**Peter Scheuermann** is a Professor of Electrical Engineering and Computer Scienc at Northwestern University. He has held visiting professor positions with the Free University of Amsterdam, the Technical University of Berlin, the Swiss Federal Institute of Technology, Zurich and University of Melbourne. Dr. Scheuermann has served on the editorial board of the Communications of ACM, The VLDB Journal, IEEE Transactions on Knowledge and Data Engineering and is currently an associate editor of Data and Knowledge Engineering, Wireless Networks and ACM Transactions on Spatial Algorithms and Systems (TSAS). Among his professional activities, he has served as General Chair of the ACM-SIGMOD Conference in 1988 and 2006, General Chair of the ER '2003 Conference and more recently as Program Co-Chair of the ACM-SIGPATIAL conference in 2009. He has published more than 140 journal and conference papers. His research has been funded by NSF, NASA, HP, Northrop Grumman and BEA, among others. Peter Scheuermann is a Fellow of IEEE and AAAS.

Please cite this article as: O.C. Ghica et al., Security of electrostatic field persistent routing: Attacks and defense mechanisms, Ad Hoc Networks (2015), http://dx.doi.org/10.1016/j.adhoc.2015.07.016